60000292

CONTROL DATA

CDC® CYBER 96X
MAINTENANCE SOFTWARE LIBRARY

REFERENCE MANUAL

MODEL DEPENDENT
CPU, MEMORY DIAGNOSTIC
TEST DESCRIPTIONS
TEST PROCEDURES

Manual History

Revision	Description	Date
A	Manual released	06-30-88

©1988 by Control Data Corporation
All rights reserved.
Printed in the United States of America.

Contents

About This Manual 5

Organization 5
Audience 5
Conventions 5
Related Publications 5
Disclaimer 6

CPU and Memory Test Descriptions 1-1

Introduction 1-1
Requirements 1-2
Characteristics 1-3

CPU and Memory Test Procedures 2-1

Introduction 2-1
Parameters and Control Words . . 2-9
Section Tests 2-34

Operator Communication 3-1

Displays 3-1
Operator Entries 3-6
Normal Messages 3-11
Error Messages 3-11
Error Codes 3-14
Applications 3-17

Glossary A-1

About This Manual

This manual describes procedures for running the tests provided on the CYBER 96X Maintenance Software Library (MSL) for use only on the CONTROL DATA® CYBER® 960 and 962 Computer Systems.

Organization

The manual defines common test procedures for CPU and Memory Tests. Detailed test program descriptions are available as separate manuals listed below.

Audience

This manual is intended for customer engineers and site personnel responsible for installing and maintaining the CYBER 960 and 962 systems.

Conventions

Because of extensive changes and reformatting of this manual, no change bars are used and all pages reflect the latest revision.

Related Publications

Control Data Publication	Publication Number
Maintenance Software Library (MSL 15X) Reference Manual	60456530
MSL 150 - Model Independent Test Descriptions	60469390
CYBER 960/962 Computer Systems Central Processor Unit (CPU) Theory of Operations/Diagrams	60000118
CYBER 960/962 CPU Hardware Maintenance Manual	60000123
CYBER Initialization Package (CIP) Reference Manual	60457180
ACT3/P - CYBER 96X MSL Test Description	60000277

Control Data Publication	Publication Number
ANT3/P - CYBER 96X MSL Test Description	60000278
BPT3/P - CYBER 96X MSL Test Description	60000279
CMT3/P - CYBER 96X MSL Test Description	60000280
CST3/P - CYBER 96X MSL Test Description	60000281
CTT3/P - CYBER 96X MSL Test Description	60000282
ICT3/P - CYBER 96X MSL Test Description	60000283
IFT3/P - CYBER 96X MSL Test Description	60000284
LMT3/P - CYBER 96X MSL Test Description	60000285
MAT3/P - CYBER 96X MSL Test Description	60000286
MCT3/P - CYBER 96X MSL Test Description	60000287
MDT3/P - CYBER 96X MSL Test Description	60000288
OIT3/P - CYBER 96X MSL Test Description	60000289
PDT3/P - CYBER 96X MSL Test Description	60000290
SMT3/P - CYBER 96X MSL Test Description	60000291

Disclaimer

This product is intended for use only as described in this document. Control Data cannot be responsible for the proper functioning of undescribed features or undefined parameters.

CPU and Memory Test Descriptions 1

Introduction 1-1

Requirements 1-2
  Hardware 1-2
  Software 1-2
  Accessories 1-2

Characteristics 1-3
  ACT3/P - Address Control Test 1-3
  ANT3/P - Arithmetic and Logic Unit (ALN) Test 1-4
  BPT3/P - Business Data Processor (BDP) Test 1-5
  CMT3/P - Memory Test 1-6
  CST3/P - Control Store Basic Paths Test 1-7
  CTT3/P - Common CST/MAC Logic Test 1-8
  ICT3/P - ICC/ICP Test 1-9
  IFT3/P - Instruction Fetch Test 1-10
  LMT3/P - Local Memory Test 1-11
  MAT3/P - Memory Array Identification Test 1-12
  MCT3/P - Maintenance Access Control Test 1-13
  MDT3/P - ALN Multiply/Divide Test 1-14
  OIT3/P - Operand Issue Test 1-15
  PDT3/P - Processor Detected Malfunctions and Associated Logic 1-16
  SMT3/P - Segment Map Test 1-17

CPU and Memory Test Descriptions 1

Introduction

This section of the manual describes procedures for running the central processing unit (CPU) fault identification tests. These tests, listed in alphabetical order, identify failures within the processor.

ACT3/P	Address Control Test
ANT3/P	Arithmetic and Logic Unit (ALN) Test
BPT3/P	Business Data Processor (BDP) Test
CMT3/P	Memory Test
CST3/P	Control Store Test (CST) Basic Paths Test
CTT3/P	Common CST/MAC Logic Test
ICT3/P	Instruction Complete Control (ICC)/Instruction Control Pipeline Test (CSP)
IFT3/P	Instruction Fetch Test
LMT3/P	Local Memory Test
MAT3/P	Memory Array Identification Test
MCT3/P	Maintenance Access Control (MAC) Test
MDT3/P	Arithmetic and Logical Network (ALN) Multiply/Divide Test
OIT3/P	Operand Issue Test
PDT3/P	ICC/ICP Processor Detected Malfunctions and Associated Logic
SMT3/P	Segment Map Test

Each identification test is divided into sections, subsections, iterations, and conditions. The conditions code manipulates the hardware under test to determine whether or not such hardware is operating correctly.

An iteration is a collection of unique condition checks that provide sufficient fault symptom data upon detecting a hardware failure so that isolation to three or less replaceable module subassemblies is possible. Iteration boundaries are normally unique patterns which the conditions use to test the hardware.

Under default parameter settings, an error stop will occur at the end of an iteration in which one or more of the iteration condition checks failed.

A subsection is a collection of iterations. Subsection boundaries normally are based on portions of a piece of a functional area.

A section is a collection of subsections. Section boundaries may be based on hardware functional boundaries or some other meaningful boundary.

To minimize the amount of time spent in developing fault identification tests, the basic test control code (BASICTC) and the diagnostic executive (DEX) are used to handle all interfaces to the common maintenance software executive (CMSE).

NOTE

CMSE must be used without the option of using central memory (CM) when running CPU identification tests. If CMSE is using central memory when these tests are run, an external fault is reported with an error code 2 (EC2) of 6057.

Requirements

This section defines the hardware and software requirements for test execution.

Hardware

Products for which test is intended:

CYBER 960 and 962 Computer Systems

Target configuration:

2 gigabytes of central memory
10 PPs
1 library device
1 display/keyboard device

Minimum configuration:

64 megabytes of central memory
5 PPs
1 library device
1 display/keyboard device

Software

All fault identification tests run under control of CMSE. The tests interface with CMSE via the DEX and the common test/diagnostic interface (CTDI) call block. The CTDI call block consists of 10 peripheral processor (PP) locations labeled TAB0 through TAB11. CMSE refers to this call block as the PP call block.

Accessories

None required.

Characteristics

The following subsections define various CPU and Memory tests.

ACT3/P - Address Control Test

1. Test name: ACT3/P
2. Test, diagnostic, utility, system: Test
3. Size (source)
   - PP code: N/A
   - Microcode: N/A
4. Size (memory required for execution)
   - PP memory: 8 K
   - Central memory: 16 MB
   - Control Store: 4 K
5. Virtual code, microcode, PP code, other: 16-bit PP code
128-bit micrands
6. Run time
   - Default sections: 60 s
   - Full test: 60 s
7. Level of isolation: Detection
8. Online, offline, both: Offline
9. Offline system: CMSE (MSL 15X)
10. Resident during execution: IOU PP
11. Languages: COMPASS 16-bit
REMEDY (microcode)
12. Source code maintenance: UPDATE
13. Uses maintenance channel: Yes

ANT3/P - Arithmetic and Logic Unit (ALN) Test

1.	Test name:	ANT3/P
2.	Test, diagnostic, utility, system:	Test
3.	Size (source)	
	– PP code:	N/A
	– Microcode:	N/A
4.	Size (memory required for execution)	
	– PP memory:	8 K
	– Central memory:	1 MB
	– Control Store:	4 K
5.	Virtual code, microcode, PP code, other:	16-bit PP code 128-bit micrands
6.	Run time	
	– Default sections:	3 min
	– Full test:	3 min
7.	Level of isolation:	Detection
8.	Online, offline, both:	Offline
9.	Offline system:	CMSE (MSL 15X)
10.	Resident during execution:	IOU PP
11.	Languages:	COMPASS 16-bit REMEDY (microcode)
12.	Source code maintenance:	UPDATE
13.	Uses maintenance channel:	Yes

BPT3/P - Business Data Processor (BDP) Test

1.	Test name:	BPT3/P
2.	Test, diagnostic, utility, system:	Test
3.	Size (source)	
	– PP code:	N/A
	– Microcode:	N/A
4.	Size (memory required for execution)	
	– PP memory:	8 K
	– Central memory:	16 MB
	– Control Store:	4 K
5.	Virtual code, microcode, PP code, other:	16-bit PP code 128-bit micrands 64-bit CP code
6.	Run time	
	– Default sections:	70 s
	– Full test:	70 s
7.	Level of isolation:	Detection
8.	Online, offline, both:	Offline
9.	Offline system:	CMSE (MSL 15X)
10.	Resident during execution:	IOU PP
11.	Languages:	COMPASS 16-bit REMEDY (microcode)
12.	Source code maintenance:	UPDATE
13.	Uses maintenance channel:	Yes

CMT3/P - Memory Test

1.	Test name	CMT3/P
2.	Test, diagnostic, utility, system:	Test
3.	Size (source)	
	– PP code:	N/A
4.	Size (memory required for execution)	
	– PP memory:	8 K
	– Central memory:	All available
5.	Virtual code, microcode, PP code, other:	16-bit PP code
6.	Run time (64-MB system)	
	– Quicklook:	4 min
	– Default sections:	9 min
	– Full test:	Not available
7.	Level of isolation:	Detection/isolation
8.	Online, offline, both:	Offline
9.	Offline system:	CMSE (MSL 15X)
10.	Resident during execution:	IOU PP
11.	Languages:	COMPASS 16-bit
12.	Source code maintenance:	UPDATE
13.	Uses maintenance channel:	Yes

CST3/P - Control Store Basic Paths Test

1. Test name: CST3/P
2. Test, diagnostic, utility, system: Test
3. Size (source)
   - PP code: N/A
   - Microcode: N/A
4. Size (memory required for execution)
   - PP memory: 8 K
   - Central memory: 16 MB
   - Control Store: 4 K
5. Virtual code, microcode, PP code, other: 16-bit PP code
128-bit micrands
6. Run time
   - Default sections: 90 s
   - Full test: 90 s
7. Level of isolation: Detection
8. Online, offline, both: Offline
9. Offline system: CMSE (MSL 15X)
10. Resident during execution: IOU PP
11. Languages: COMPASS 16-bit
REMEDY (microcode)
12. Source code maintenance: UPDATE
13. Uses maintenance channel: Yes

CTT3/P - Common CST/MAC Logic Test

1.	Test name:	CTT3/P
2.	Test, diagnostic, utility, system:	Test
3.	Size (source)	
	– PP code:	N/A
	– Microcode:	N/A
4.	Size (memory required for execution)	
	– PP memory:	8 K
	– Central memory:	16 MB
	– Control Store:	4 K
5.	Virtual code, microcode, PP code, other:	16-bit PP code Microcode
6.	Run time	
	– Default sections:	22 s
	– Full test:	22 s
7.	Level of isolation:	Detection
8.	Online, offline, both:	Offline
9.	Offline system:	CMSE (MSL 15X)
10.	Resident during execution:	IOU PP
11.	Languages:	COMPASS 16-bit REMEDY (microcode)
12.	Source code maintenance:	UPDATE
13.	Uses maintenance channel:	Yes

ICT3/P - ICC/ICP Test

1.	Test name:	ICT3/P
2.	Test, diagnostic, utility, system:	Test
3.	Size (source)	
	– PP code:	N/A
	– Microcode:	N/A
4.	Size (memory required for execution)	
	– PP memory:	8 K
	– Central memory:	16 MB
	– Control Store:	4 K
5.	Virtual code, microcode, PP code, other:	16-bit PP code Microcode
6.	Run time	
	– Default sections:	31 s
	– Full test:	31 s
7.	Level of isolation:	Detection
8.	Online, offline, both:	Offline
9.	Offline system:	CMSE (MSL 15X)
10.	Resident during execution:	IOU PP
11.	Languages:	COMPASS 16-bit REMEDY (microcode)
12.	Source code maintenance:	UPDATE
13.	Uses maintenance channel:	Yes

IFT3/P - Instruction Fetch Test

1. Test name:	IFT3/P
2. Test, diagnostic, utility, system:	Test
3. Size (source):	N/A
4. Size (memory required for execution)	
– PP memory:	8 K
– Central memory:	16 MB
– Control Store:	4 K
5. Virtual code, microcode, PP code, other:	16-bit PP code Microcode
6. Run time	
– Quicklook:	45 s
– Default sections:	50 s
– Full test:	50 s
7. Level of isolation:	Detection
8. Online/offline/both:	Offline
9. Offline system:	CMSE (MSL 15X)
10. Resident during execution:	IOU PP
11. Languages:	COMPASS 16-bit REMEDY (microcode)
12. Source code maintenance:	UPDATE
13. Uses maintenance channel:	Yes

LMT3/P - Local Memory Test

1.	Test name:	LMT3/P
2.	Test, diagnostic, utility, system:	Test
3.	Size (source):	
	– PP code:	N/A
	– Microcode:	N/A
4.	Size (memory required for execution)	
	– PP memory:	8 K
	– Central memory:	16 MB
	– Control Store:	4 K
5.	Virtual code, microcode, PP code, other:	16-bit PP code 128-bit micrands
6.	Run time	
	– Default sections:	6 min
	– Full test:	6 min
7.	Level of isolation:	Detection
8.	Online, offline, both:	Offline
9.	Offline system:	CMSE (MSL 15X)
10.	Resident during execution:	IOU PP
11.	Languages:	COMPASS 16-bit REMEDY (microcode)
12.	Source code maintenance:	UPDATE
13.	Uses maintenance channel:	Yes

MAT3/P - Memory Array Identification Test

1.	Test name:	MAT3/P
2.	Size (source lines)	
	- PP code:	N/A
	- Microcode:	N/A
3.	Size (memory required for execution)	
	- PP memory:	8 K
	- Central memory:	All available
	- Control Store:	4 K
4.	Virtual code, microcode, PP code, other:	16-bit PP code REMEDY (microcode)
5.	Online, offline, both:	Offline
6.	Offline system:	CMSE-180
7.	Resident during execution:	IOU PP
8.	Languages:	COMPASS 16-bit REMEDY (microcode)
9.	Source code maintenance:	UPDATE

MCT3/P - Maintenance Access Control Test

1. Test name:	MCT3/P
2. Test, diagnostic, utility, system:	Test
3. Size (source)	
– PP code:	N/A
4. Size (memory required for execution)	
– PP memory:	8 K
– Control Store:	4 K
5. Virtual code, microcode, PP code, other:	16-bit PP code
6. Run time	
– Default sections:	90 s
– Full test:	90 s
7. Level of isolation:	Detection
8. Online/offline/both:	Offline
9. Offline system:	CMSE (MSL 15X)
10. Resident during execution:	IOU PP
11. Languages:	COMPASS 16-bit
12. Source code maintenance:	UPDATE
13. Uses maintenance channel:	Yes

MDT3/P - ALN Multiply/Divide Test

1.	Test name:	MDT3/P
2.	Test, diagnostic, utility, system:	Test
3.	Size (source)	
	– PP code:	N/A
	– Microcode:	N/A
4.	Size (memory required for execution)	
	– PP memory:	8 K
	– Control Store:	4 K
5.	Virtual code, microcode, PP code, other:	16-bit PP code Microcode
6.	Run time	
	– Default sections:	3 min
	– Full test:	3 min
7.	Level of isolation:	Detection
8.	Online, offline, both:	Offline
9.	Offline system:	CMSE (MSL 15X)
10.	Resident during execution:	IOU PP
11.	Languages:	COMPASS 16-bit REMEDY (microcode)
12.	Source code maintenance:	UPDATE
13.	Uses maintenance channel:	Yes

OIT3/P - Operand Issue Test

1.	Test name:	OIT3/P
2.	Test, diagnostic, utility, system:	Test
3.	Size (source)	
	- PP code:	N/A
4.	Size (memory required for execution)	
	- PP memory:	8 K
	- Central memory:	16 MB
	- Control Store:	4 K
5.	Virtual code, microcode, PP code, other:	16-bit PP code 128-bit micrands
6.	Run time	
	- Default sections:	50 s
	- Full test:	50 s
7.	Level of isolation:	Detection
8.	Online/offline/both:	Offline
9.	Offline system:	CMSE (MSL 15X)
10.	Resident during execution:	IOU PP
11.	Languages:	COMPASS 16-bit REMEDY (microcode)
12.	Source code maintenance:	UPDATE
13.	Uses maintenance channel:	Yes

PDT3/P - Processor Detected Malfunctions and Associated Logic

1. Test name: PDT3/P
2. Test, diagnostic, utility, system: Test
3. Size (source)
   - PP code: N/A
   - Microcode: N/A
4. Size (memory required for execution)
   - PP memory: 8 K
   - Central memory: 16 MB
   - Control Store: 4 K
5. Virtual code, microcode, PP code, other: 16-bit PP code
128-bit micrands
6. Run time
   - Default sections: 18 s
   - Full test: 18 s
7. Level of isolation: Detection
8. Online, offline, both: Offline
9. Offline system: CMSE (MSL 15X)
10. Resident during execution: IOU PP
11. Languages: COMPASS 16-bit
REMEDY (microcode)
12. Source code maintenance: UPDATE
13. Uses maintenance channel: Yes

SMT3/P - Segment Map Test

1.	Test name:	SMT3/P
2.	Test, diagnostic, utility, system:	Test
3.	Size (source)	
	– PP code:	N/A
	– N/A	N/A
	– Microcode:	
	– N/A	
4.	Size (memory required for execution)	8 K
	– PP memory:	16 MB
	– Central memory:	4 K
	– Control Store:	
5.	Virtual code, microcode, PP code, other:	16-bit PP code Microcode
6.	Run time	
	– Default sections:	47 s
	– Full test:	47 s
7.	Level of isolation:	Detection
8.	Online, offline, both:	Offline
9.	Offline system:	CMSE (MSL 15X)
10.	Resident during execution:	IOU PP
11.	Languages:	COMPASS 16-bit REMEDY (microcode)
12.	Source code maintenance:	UPDATE
13.	Uses maintenance channel:	Yes

CPU and Memory Test Procedures 2

Introduction 2-1
  Order of Execution 2-1
  Logic Not Being Tested 2-1
  Clock Pulse Width Margins 2-7
    MCT3/P 2-7
  Real Memory Address Limitations 2-7
    ACT3/P and PDT3/P 2-7
  Cache and Page Map Set Availability 2-7
    LMT3/P 2-7
    Loading Procedure 2-7

Parameters and Control Words 2-9
  ACT3/P 2-12
  ANT3/P 2-13
  BPT3/P 2-14
  CMT3/P 2-15
  CST3/P 2-18
  CTT3/P 2-19
  ICT3/P 2-20
  IFT3/P 2-21
  LMT3/P 2-22
  MAT3 2-23
  MCT3/P 2-25
  MDT3/P 2-26
  OIT3/P 2-27
  PDT3/P 2-28
  SMT3/P 2-29
  Control Words 2-30
  Quicklook Section Selection 2-31
  Test Error Buffer 2-31
  Expected Result Buffer 2-33

Section Tests 2-34
  ACT3/P - Address Control Test 2-34
  ANT3/P - Arithmetic and Logic Unit (ALN) Test 2-37
  BPT3/P - Business Data Processor (BDP) Tests 2-45
  CMT3/P - Memory Test 2-51
  CST3/P - Control Store Basic Paths Test 2-56
  CTT3/P - Common CST/MAC Logic Test 2-61
  ICT3/P - Instruction Complete Control, Instruction Control Pipeline Test 2-64
  IFT3/P - Instruction Fetch Test 2-67
  LMT3/P - Local Memory Test 2-69
  MAT3/P - Memory Array Identification Test 2-77
  MCT3/P - MAC Test 2-78
  MDT3/P - ALN Multiply/Divide Test 2-87
  OIT3/P - Operand Issue Test 2-93
  PDT3/P - Processor Detected Malfunction and Associated Logic 2-97
  SMT3/P - Segment Map Test 2-99

CPU and Memory Test Procedures 2

Introduction

The following paragraphs explain the operational procedures related to the central processing unit (CPU) and Memory tests. Refer to individual test descriptions for restrictions and user cautions.

Order of Execution

CPU and Memory fault identification tests should be run in the following sequence:

CMT3/P
MCT3/P
CST3/P
IFT3/P
CTT3/P
OIT3/P
ACT3/P
LMT3/P
SMT3/P
ANT3/P
MDT3/P
BPT3/P
PDT3/P
ICT3/P
MAT3/P

Logic Not Being Tested

Address Control logic not tested by ACT3/P includes several areas within Address Control used exclusively by other functional blocks. These areas will be tested by the diagnostic responsible for that block, and must follow the execution of ACT3/P.

Logic not tested by ACT3/P but located within Address Control is used by:

- BDP to store, load, and generally control the flow of Business Data Processing (BDP) stream data. This logic will be tested by BPT3/P.
- Arithmetic and Logic Network (ALN) to provide a shift count. This logic will be tested by ANT3/P.
- Local Memory (LM) to append the Active Segment Identifier (ASID) to the Byte Address. This logic will be tested by LMT3/P.

Arithmetic and logic network logic not tested by ANT3:

- CST3/P tests the RDSB/ADATB data path through the Multiply/Divide unit into bits 64 through 127 of the Shift Mux and output through the Output Mux from Shifter bits 64 through 127 to Data Interchange (DAI) and the Register File.
- CST3/P also tests the RDSC/ADATC data path to the Shift Mux and output through the Output Mux from Shifter bits 64 through 127 to DAI and the Register File.
- MDT3/P tests all output signals from the Multiply/Divide unit except for the RDSB/ADATB data path and C sign selection logic in the Multiply/Divide unit. ANT3/P assumes that all Multiply/Divide units signals to the general network are inactive.
- MDT3/P also tests all micrand controls that decode to drive the Multiply/Divide unit or sample outputs from the Multiply/Divide unit.
- PDT3/P and ICT3/P test all Processor Detected Malfunction (PDM) indications.
- CST3/P tests the ALN Soft Control Memory RAMs from the Maintenance Channel (MCH).
- MDT3/P tests the ALN Soft Control Memory RAMs addressing and output to Multiply/Divide unit.
- BPT3/P tests ALN Micrand Aux no-response logic.

Memory logic not tested by CMT3/P:

- Port logic not physically connected to the PP. This will have to be tested by tests executing from the equipment connected to the particular port being tested. IFT3/P, CST3/P, and LMT3/P identification tests check some of this logic.
- The Free Run Counter. This maintenance register will be tested by LMT3/P.
- Partial write select logic. This logic is tested by the ACT3/P identification test.
- Interrupt. More than one PP will be needed to verify operation of the interrupt system.
- Bank busy logic is tested by LMT3/P.
- Bank conflict and priority logic is tested by LMT3/P.

Control Store logic not tested by CST3/P includes some areas within Control Store which are tested by other identification tests.

Logic in Control Store not tested by CST3/P:

- General Micrand RAM and FU Micrand RAMs; along with the General Micrand Register, FU Micrand Registers, Control Store Disassembly Register, Micrand Sequence Control Register, Micrand Address Register, Breakpoint Register, Micrand Address Incrementor, and related parity checkers are tested by MCT3/P.
- Debug control logic will be tested by CTT3/P.
- MAC Microcode Assist Control Store logic will be tested by CTT3/P.
- Exit signal input to Rank 125 empty is tested by IFT3/P.

Control Store logic not tested by CTT3/P includes all logic areas of Control Store tested by MCT3/P and CST3/P. CTT3/P is intended to test the few remaining areas of Control Store logic and MAC logic that could not be tested by MCT3/P and CST3/P.

ICC logic not tested by ICT3/P includes portions of the ICC logic checked by tests other than ICT3/P:

- CST3/P tests Deadman Timeout operation, the data path between the maintenance channel and the User Condition Register (UCR) and Monitor Condition Register (MCR), and the trap, exchange, and halt interrupts generated by Deadman Timeouts.
- LMT3/P tests that an external interrupt from the IOU sets the Processor Halted Flip Flop and MCR bit 8.
- PDT3/P tests the setting of all Rank 50 bits in the User Condition Register (UCR) and the Monitor Condition Register (MCR).
- ANT3/P tests the setting and clearing of the Divide Fault Flip-Flop (FF) and that the FF can disable Register File writes. It tests the setting of UCR bits 7, 9, 10, 11, 12, 13, and 14, and partially tests the ability of ICC to interrupt when these bits are set.

Logic not tested by IFT3/P includes some Instruction Fetch logic. To test this logic would involve untested logic outside of the IF area. Untested logic:

- The unbranch signal (UNBR) from ALN is tested by ANT3/P.
- The Branch Address Adder input from ICP rank 50 is tested by ANT3/P.
- Page table search without Find (PSWFIF) from Local Memory is tested by LMT3/P.
- The enable unbranch signal (ENUNBR) from ICP is tested by ANT3/P.
- The not buffer clear signal (NBFCLR) going to CST is tested by ANT3/P.
- The branch not taken signal (NUNBR) going to ICC is tested by ANT3/P.
- The branch not taken logic in IF is tested by ANT3/P.
- Bits 37 through 39 of C170 RAC input is tested by ACT3/P.
- IF PDM logic will be tested by PDT3/P.

Local memory logic not tested by LMT3/P includes several logic areas within local memory used by other processor functional blocks. These areas must be tested by the diagnostic responsible for the functional block.

- SMT3/P tests the Instruction Issue Request enable (LMNMIS) when using Process Virtual Address (PVA) to Real Memory Address (RMA) conversion.
- SMT3/P also tests the Purge Code parity signal (LMCPE 0) to MAC.
- BPT3/P tests all logic associated with the Address Control B-Stream; that is, B-Stream request, response, and Tag File logic.
- ACT3/P tests the Mark Bits parity checker.
- ACT3/P also tests the basic Real Memory Address (RMA) address path and the CPU to CMC data path.
- OIT3/P tests the Exchange Jump Word Number (IITGO) parity checker.
- ICT3/P and PDT3/P tests all Processor Detected Malfunction (PDM) indications.

Memory logic not tested by MAT3/P includes all logic in memory tested by CMT3/P. This logic is external to the memory array boards. CMT3/P also tests all data, address, and control lines into the memory array boards.

MAC logic not tested by MCT3/P includes some areas within the MAC logic tested by other identification tests. These other identification tests must be executed after MCT3/P is executed, excluding CMT3/P which would normally be executed before the tests.

Logic in MAC that is not tested by MCT3/P:

- Logic used for MAC reading and writing of processor registers which require microcode assistance. This logic will be tested by CTT3/P.
- Logic which connects the MAC to the memory maintenance register. This logic will be tested by memory CMT3/P.
- Logic used for MAC reading and writing of the processor Performance Monitor registers. The Performance Monitor (PMF) is a customer option. This logic in MAC will be tested by PMT3/P.
- Logic used for ECS operations. This logic must be tested as part of a ECS test.
- Exchange Accept, Status Summary OR, Uncorrected Error, and Corrected Error logic will all be tested by ICT3/P.
- Individual bits in the processor PFS Registers (which indicate parity errors from individual areas of the processor) cannot all be verified as operational by MCT3/P, since many areas of the processor cannot be exercised by MCT3/P. The identification tests executed after MCT3/P will ensure the operation of PFS bits not verified by MCT3/P.

Multiply/divide unit logic not tested by MDT3/P:

- CST3/P tests the following logic in the Multiply/Divide unit:
  - Multiply/Divide C Register loading of C Operand data.
  - Multiply/Divide Output Mux for selecting of B Operand data or C Register Data.
- MCT3/P tests the ALN Soft Control Memory RAMs from the maintenance channel.
- ANT3/P tests the Multiply/Divide unit B and C Sign Fanout Logic.
- BPT3/P tests the BDP Convert Bits Register and its input to the Multiply Partial Summing Network.

Operand issue logic not tested by OIT3/P includes several areas within Operand Issue controlled by other functional blocks. These areas are tested by diagnostics responsible for those blocks. Logic located within Operand Issue but not tested by OIT3/P:

- The Ref ROM Address (Live) Register and Register File including the Register File access from MAC, which are tested by MCT3/P.
- The Segment Table Address and Segment Table Length (Live) Registers and the logic used to perform segment descriptor fetches, which are tested by SMT3/P.
- The Exchange Address and Error Condition (Live) Registers, the logic used to treat Hold, Register File Shortstop, Operand Shortstop conditions, and the associated conflict checks, which are tested by ICT3/P.
- The BDP Condition (Live) Register, which is tested by BPT3/P.
- The Page Size Mask (Live) Register, DAI Error Path Decoder, LM tag bits/tag address and parity check, which are tested by LMT3/P.
- The streaming operation used during exchanges, which is tested by LMT3/P and ICT3/P.
- The logic that detects and resolves operand conflicts and is tested by ICT3/P.

ICC/ICP logic not tested by PDT3/P includes all logic checked by test ICT3/P. This includes logic such as trap handling, disabling register writes, and aborting of operations. PDT3/P primarily tests logic that is needed to set various condition registers in ICC.

Segment map logic not tested by SMT3/P (with the test that tests the logic shown in parentheses):

- IF ASID to LM (LMT3/P).
- SM Bypass Address Mux (CST3/P, LMT3/P).
- Initial/Final ring compare (CST3/P).
- Purge code to LM (LMT3/P).
- Access Violation or Invalid Segment signal to Address Control (ACT3/P).
- Access Violation or Invalid Segment signal to LM (LMT3/P).

Clock Pulse Width Margins

MCT3/P

Because MCT3/P tests the DEC register in the early test sections, clock pulse width margins are not enabled until section 6.

Real Memory Address Limitations

ACT3/P and PDT3/P

The RMA used to test address validity will be limited to the minimum configuration, one million bytes (1 MB), or expressed in words, 131 K.

Cache and Page Map Set Availability

LMT3/P

LMT3/P assumes all three Page Map sets are available for use.

LMT3/P also assumes all Cache sets configured are available. Any attempt to reduce the number of available sets, because of known hardware deficiencies, by presetting the processor Environment Control Register is ignored.

Loading Procedure

All CPU and memory fault identification tests are loaded in the same way.

The identification tests are PP programs that are loaded into a PP using the CMSE load PP program command:

```
CP,num,test name (CR)
```

where num is the PP number.

The test can be loaded and put into execution with the command:

```
LT,num,test name (CR)
```

Prior to loading the test into a PP, the PP must have been deadstarted using the CMSE command:

```
DP,num (CR)
```

It is expected that in most cases, the memory identification test will execute in PP 04.

A command buffer exists on the CIP 960 or CIP 962 tape to facilitate loading and execution of each test. Refer to the following list for the names of command buffers provided. Before using a command buffer, you may have to modify it for your system. Display the command buffer using the CMSE command buffer display commands, or print the contents of the command buffer using procedures provided in the Command Buffer Maintenance section of the MSL 15X Reference manual listed in About This Manual. Modify the command buffer as directed by comments embedded in the command buffer. When you are satisfied that the command buffer is set up properly, enter a GO, xxx3 command (where xxx3 is the command buffer name) to execute it.

Modified command buffers can be saved on a backup tape. Refer to the Utilities section in the MSL 15X Reference manual listed in About This Manual for procedures.

Names of command buffers provided for running individual tests are the same as the test name. Two additional command buffers are provided.

- ALLMD3 which runs all xxT3/P tests in sequential order.
- ALLMD3B which runs all xxT3/P tests in sequential order on CPU-1 of a two CPU system.

Command buffer and test names are cross-referenced in the following list.

Command Buffer	Test
CMT3	CMT3/P
MCT3	MCT3/P
CST3	CST3/P
IFT3	IFT3/P
CTT3	CTT3/P
OIT3	OIT3/P
ACT3	ACT3/P
LMT3	LMT3/P
SMT3	SMT3/P
ANT3	ANT3/P
MDT3	MDT3/P
BPT3	BPT3/P
PDT3	PDT3/P
ICT3	ICT3/P
MAT3	PMT3/P
ALLMD3	Runs all of above in sequential order.
ALLMD3B	Runs same sequence as ALLMD3 on CPU-1 of a two CPU system.

Parameters and Control Words

The following parameters provide control information during execution of the identification tests.

Tag	Octal Address	Bit(s)	Meaning
PARAM0	122	2^0	Stop at end of test (ST).
		2^1	Stop at end of section (SS).
		2^2	Stop at end of subsection (SB).
		2^3	Stop at end of condition (SC).
		2^4	Stop on error (SE).
		2^5	Log errors in dayfile (LE).
		2^6	Repeat test (RT).
		2^7	Repeat section (RS).
		2^8	Repeat subsection (RB).
		2^9	Repeat condition (RC).
		2^{10}	Scope mode (SM).
		2^{11}	Quicklook flag (QL).
		2^{12}	Bypass all stops and messages (DR). Overrides bit 2^{13}.
		2^{13}	Display only error messages (DE). Bypasses all stops and messages except error stops and error messages.
		2^{14}	Reserved for CMSE use.
		2^{15}	Enable CMSE to alter PARAM0 bits via the Cxx/Sxx commands.
			Default selection is $100021_8/8011_{16}$. The keyboard commands to set and clear these bits are described in the MSL 15X Reference manual.
PARAM1	123	2^0	Select wide clock margins only in CPU for all xxT3 tests except CMT3/P which selects wide clock margins only in memory.
		2^1	Select narrow clock margins only in CPU for all xxT3 tests except CMT3/P which selects narrow clock margins only in memory.
		2^2	Bypass parameter stop.
		2^3	Not used.
		2^4	Not used.
		2^5	Not used.
		2^6	Isolate failure to replaceable subassembly.[1]
		2^7	Not used.
		2^8	Report condition errors (CE).
		2^9	Bypass error code 1 cracker.
		2^{10} and 2^{11}	Not used (zeros).
		2^{12}	Stop at end of iteration (SI). Will stop after each pass of a multi-iteration sequence.

---

[1] Available for CMT3/P only.

Tag	Octal Address	Bit(s)	Meaning
		2^{13}	Repeat iteration (RI). Repeat only the current iteration (pattern).
		2^{14}	Abort subsection on error (AB).
		2^{15}	Tell CMSE to process AB, RI, CI, AN, CE commands in program 1. Default is 1.
PARAM2	124	2^0 through 2^{15}	Repeat test count. The test will repeat the number of times contained in this location, minus one (for example, PARAM2 equals 0002 will cause the test to repeat once, resulting in two passes through the test). The test will then unconditionally exit if repeat test parameter bit, PARAM0 2^6, is not set. Default selection is zero.
PARAM3	125	2^0 through 2^3	x
		2^4 through 2^{15}	yyy Stop at end of condition x when number of iterations equals yyy. The first iteration is iteration zero. This parameter is used in subsections containing multiple conditions and iterations. It allows the user to obtain a failing condition and iteration without stepping through the subsection. A nonzero value in PARAM3 is required to activate this process. Default selection is zero.
PARAM4	126	2^0 through 2^7	Reserved, forced to zero.
		2^8 through 2^{11}	The maintenance channel connect code for the hardware unit to be tested. Default selection is one.
		2^{12} through 2^{15}	Reserved, forced to zero.
PARAM5	127	2^0 through 2^{15}	Select sections 00 through 15. Refer to Test-Defined Parameters.
PARAM6	130	2^0 through 2^{10}	Select sections 16 through 31. Refer to Test-Defined Parameters.
PARAM7	131	2^0 through 2^{15}	Select sections 32 through 47. Refer to Test-Defined Parameters.
PARAM8	132	2^0 through 2^{15}	Select sections 48 through 63. Refer to Test-Defined Parameters.

Tag	Octal Address	Bit(s)	Meaning
PARAM9	133	2^0 through 2^{15}	Test-defined. Refer to Test-Defined Parameters.
PARAM10	134	2^0 through 2^{15}	Test-defined. Refer to Test-Defined Parameters.
PARAM11	135	2^0 through 2^{15}	Test-defined. Refer to Test-Defined Parameters.
PARAM12	136	2^0 through 2^{15}	Test-defined. Refer to Test-Defined Parameters.
PARAM13	137	2^0 through 2^{15}	Test-defined. Refer to Test-Defined Parameters.
PARAM14	140		Test-defined.
PARAM15	141		Test-defined.
PARAM16	142	2^0	Test-defined. Refer to Test-Defined Parameters. Refer to Restrictions and User Cautions.
		2^1 through 2^{15}	Not used.
PARAM17	143	2^0 through 2^{15}	Not used.
PARAM18	144		Repeat subsection count. The test will automatically repeat each subsection in the selected sections the number of times contained in this location, minus one; (for example, PARAM18 equals 0002 will cause each subsection to be repeated once, resulting in two passes through each subsection). Repeat subsection PARAM0 bit 2^8 must be clear.

ACT3/P

The following parameters are used by the ACT3/P identification test.

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{15}	Select sections 00 through 15.
PARAM6	130	2^0 through 2^{10}	Select sections 16 through 26.
		2^{11} through 2^{15}	Not used.
PARAM7	131	2^0 through 2^{15}	Not used.
PARAM8	132	2^0 through 2^{15}	Not used.
PARAM9 through PARAM15	133 through 141		Not used.
PARAM16	142	2^0	When zero, allows margins to be run per PARAM1, bits 2^0 and 2^1.
			2^1 through 2^{15} Not used.

When Quicklook is selected (PARAM0 bit 2^{11}), sections 00 through 03 and 05 through 26 will be run.

ANT3/P

The following parameters are used by the ANT3/P identification test.

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{15}	Select sections 00 through 15.
PARAM6	130	2^0 through 2^{12}	Select sections 16 through 28.
		2^{13} through 2^{15}	Not used.
PARAM7	131	2^0through 2^{15}	Not used.
PARAM8	132	2^0 through 2^{15}	Not used.

BPT3/P

The following parameters are used by the BPT3/P identification test.

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{15}	Select sections 00 through 15.
PARAM6	130	2^0 through 2^{15}	Select sections 16 through 31.
PARAM7	131	2^0 through 2^7	Select sections 32 through 39.
		2^8 through 2^{15}	Not used.
PARAM8	132	2^0 through 2^{15}	Not used.
PARAM9 through PARAM15	133 through 141		Not used.
PARAM16	142	2^0	When zero, allows margins to be run per PARAM1, bits 2^0 and 2^1.
		2^1 through 2^{15}	Not used.

When Quicklook is selected, sections 00 through 39 will be run.

CMT3/P

NOTE

The following CMT3/P test parameters are system-dependent:

PARAM4	MCH/Element connect code. (Refer to Parameters and Control Words).
PARAM12	Memory port connected to IOU.
PARAM13	Memory installed.
PARAM14	Memory serial number.

The following parameters are used by the CMT3/P identification test.

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{15}	Select sections 00 through 15. Default selection is 177777_8.
PARAM6	130	2^0 through 2^{11}	Select sections 16 through 28. Default selection is 11577_8.
		2^{12} through 2^{15}	Not used.
PARAM7	131	2^0 through 2^{15}	Not used.
PARAM8	132	2^0 through 2^{15}	Not used. **NOTE** Quicklook section selects are 0 through 17, 24 through 25.
PARAM9	133	2^0 through 2^{15}	Contains a threshold count representing the number of single-bit errors allowed on a single memory chip. Default is zero.
PARAM10	134	2^0 through 2^{14}	Contains a definition of the first word address used in array testing. The test forms first word address by appending 14 zero bits to the contents of parameter 10. This parameter does not prohibit use of any address by the test. (Default select is 0.) (Refer to PARAM11 for entry format.)
		2^{15}	Not used.
PARAM11	135	2^0 through 2^{15}	Contains bits 2^{14} through 2^{28} of the last word address plus one. This parameter defines the upper limit of the array tests. It does not prohibit use of any address by the test. This value will automatically be adjusted if the amount of physical memory is less. (Default selection is 20_8.) **NOTE** The last address referenced will be one less than the address portion entered.
		2^{15}	Set if last address used is to be the last physical CM address as specified by PARAM13.

Tag	Octal Address	Bit(s)	Meaning
PARAM12	136	2^0 through 2^1	Memory port number.
		2^2	Enable Refresh disable switch.
		2^3 through 2^{15}	Not used. Default is 1.
PARAM13	137		Options installed. Default is 0.
		2^0 through 2^3	Not used.
		2^4	1 MB
		2^5	2 MB
		2^6	4 MB
		2^7	8 MB
		2^8	16 MB
		2^9	32 MB
		2^{10}	64 MB
		2^{11}	128 MB
		2^{12}	256 MB
		2^{13}	512 MB
		2^{14}	1 GB
		2^{15}	2 GB
PARAM14	140	2^0 through 2^{15}	Memory serial number in hexadecimal digits. The decimal value is entered as though it is hexadecimal. Refer to bytes six and seven of Element Identification Register. If zero, test will not report differences. Default is 0.
PARAM15	141		Not used.
PARAM16	142	$2^0 = 1$	CMT3/P retains margin settings in DEC registers.
		$2^0 = 0$	BASICTC sets margins as specified in PARAM1.
		2^1 through 2^{15}	Not used. Default selection is 1.

CST3/P

The following parameters are used by the CST3/P identification test.

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{12}	Select sections 00 through 12.
		2^{13} through 2^{15}	Not used.
PARAM6 through PARAM15	130 through 141		Not used.
PARAM16	142	2^0	When 0, allow Clock Pulse Width margins to be run per PARAM1, bits 2^0 and 2^1.
		2^1 through 2^{15}	Not used.

There is no difference for CST3/P when run with default section selects than when run with Quicklook selected. Sections 0 through 12 will be run when the Quicklook parameter bit is set.

CTT3/P

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^5	Select sections 0 through 5.
		2^6 through 2^{12}	Not used.
PARAM6 through PARAM15	130 through 141		Not used.

When Quicklook is selected, sections 0 through 5 will be run.

ICT3/P

Tag	Octal Address	Default Value	Bit(s)	Meaning
PARAM5	127	017777	2^0 through 2^{12}	Select test sections 00 through 12.
			2^{13} through 2^{15}	Not Used.
PARAM6 through PARAM8	130 through 132	0	2^0 through 2^{15}	Not used.
PARAM9	133	100000	2^0 through2^4	If bit 2^{15} is clear, these bits contain the number of the PP which the test will use as a monitor PP during test execution. If bit 2^{15} is set, these bits are not used.
			2^5 through 2^{14}	Not used.
			2^{15}	If set, allows CMSE to automatically select the first available PP as the monitor PP used by this test. If clear, the PP number of the monitor PP is obtained directly from bits 2^0 through 2^4.
PARAM10	134	100000	2^0 through 2^4	If bit 2^{15} is clear, these bits contain the channel number to be used as communications channel between Test PP and Monitor PP. If bit 2^{15} is set, these bits are not used.
			2^5 through 2^{14}	Not used.
			2^{15}	If bit 2^{15} is set, the channel number is the same as the PP number of the monitor PP (as determined by the contents of PARAM9). If clear, the channel number is obtained directly from bits 2^4 through 2^0.
PARAM11	135	0	2^0 through 2^{15}	Reserved.
PARAM12 through PARAM15	136 through 141	0	2^0 through 2^{15}	Not used.

IFT3/P

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{15}	Select sections 0 through 9. Default is 1777_8.
PARAM6	130	2^0 through 2^{15}	Not used.
PARAM7	131	2^0 through 2^{15}	Not used.
PARAM8	132	2^0 through 2^{15}	Not used.

NOTE

Quicklook section selects are sections 0 through 9. (QL Limits operand sets in Sections 0 and 1.)

Tag	Octal Address	Bit(s)	Meaning
PARAM16	142	2^0 through 2^{15}	Bypass BASICTC setting of clock width margins. Default is 0.

LMT3/P

Cache sets available.

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{15}	Select sections 00 through 15. Default value is $FFFF_{16}$.
PARAM6	130	2^0 through 2^{15}	Select sections 16 through 31. Default value is $FFFF_{16}$.
PARAM7	131	2^0 through 2^{15}	Select sections 32 through 47. Default value is $FFFF_{16}$.
PARAM8	132	2^0 through 2^3	Select sections 48 through 51. Default value is $000F_{16}$.
		2^4 through 2^{15}	Not used.

NOTE

Quicklook section selects are sections 0 through 51.

Tag	Octal Address	Bit(s)	Meaning
PARAM9	133	2^0 through 2^{15}	Not used.
PARAM10	134	2^0 through 2^{15}	Not used.
PARAM11	135	2^0 through 2^{15}	Not used.
PARAM12	136	2^0 through 2^{15}	Not used.
PARAM13	137	2^0 through 2^{15}	Not used.
PARAM14	140	2^0 through 2^{15}	Not used.
PARAM15	141	2^0 through 2^{15}	Not used.
PARAM16	142	2^0	Allows margins to be run as defined by PARAM1, bits 2^0 and 2^1.
		2^1 through 2^{15}	Not used.

MAT3

Tag	Octal Address	Default Value (Octal)	Bit(s)	Meaning
PARAM5	127	77	2**0 through 2**5	Section selects 0 through 5.
			2**6 through 2**15	Not used.
PARAM6 through PARAM8	130 through 132	0	2**0 through 2**15	Not used.
PARAM9	133	4	2**0 through 2**14	If bit 2**15 is clear, these bits define the number of iterations of subsection 5.0 which are to be run.
			2**15	If set, subsection 5.0 shall run indefinitely, so that the test will only halt on an error. If clear, the number of iterations of subsection 5.0 to be run is obtained from bits 2**0 through 2**14.
PARAM10	134	0	2**0 through 2**15	Defines the lowest CM word address to be tested (inclusive). Address is formed as the contents of PARAM10, appended on the right with 10 bits of zeros.
PARAM11	135	100000	2**0 through 2**14	If bit 2**15 is clear, these bits define the highest CM word address to be tested (inclusive). Address is formed as the contents of these bits, appended on the right with 10 bits of ones.
			2**15	If set, the highest CM word address to be tested is the highest physical memory address (as specified by the contents of the Options Installed Rgtr.). If clear, the highest CM word address to be tested is obtained from bits 2**0 through 2**14.

Tag	Octal Address	Default Value (Octal)	Bit(s)	Meaning
PARAM12 through PARAM13	136 through 137	177777	2**0 through 2**15	If PARAM12 and PARAM13 both equal $FFFF_{16}$, the Random Number Generator offset value is formed from the contents of the Free Running Counter, as captured during test initialization.. If either PARAM12 or PARAM13 is not equal to $FFFF_{16}$, the Random Number Generator offset value is formed by copying the contents of PARAM12 into bytes 0 through 1 and 4 through 5, and the contents of PARAM13 into bytes 2 through 3 and 6 through 7.
PARAM14	140	0	2**0 through 2**15	Reserved.
PARAM15	141	0	2**0 through 2**15	Not used.

MCT3/P

The following parameters are used by the MCT3/P identification test.

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{15}	Select sections 00 through 15.
PARAM6	130	2^0 through 2^{12}	Select sections 16 through 28.
		2^{13} through 2^{15}	Not used.
PARAM7	131	2^0 through 2^{15}	Not used.
PARAM8	132	2^0 through 2^{15}	Not used.
PARAM9	133	2^0 through 2^{15}	Processor serial number in hexadecimal.
PARAM10	134	2^0 through 2^2	Options installed: 2^0 set Performance Monitor installed. 2^1 set Cache Expansion installed. 2^2 set PMF/ECS Interface installed.
		2^3 through 2^7	Must be clear.
		2^8	If set, processor is not in central memory control (CMC) cabinet. If clear, processor is in CMC cabinet.
		2^9 through 2^{15}	Must be clear.
PARAM11 through PARAM15	135 through 141	Not used.	
PARAM16	142	2^0	If set, prevents BASICTC from changing processor clock pulse width margins. Section 6 will change margins according to PARAM1 bits 2^0 and 2^1 selection. Default is 1.
		2^1 through 2^{15}	Not used.

When Quicklook is selected (PARAM0 bit 2^{11}), sections 00 through 28 will be run.

MDT3/P

The following parameters are used by the MDT3/P identification test.

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{11}	Select sections 00 through 11.
		2^{12} through 2^{15}	Not used.
PARAM6	130	2^0 through 2^{12}	Not used.
PARAM7	131	2^0 through 2^{15}	Not used.
PARAM8	132	2^0 through 2^{15}	Not used.
PARAM16	142	2^0	When equal to 0, allow clock pulse width margins to be run according to PARAM1 bit 2^1.
		2^1 through 2^{15}	Not used.
Quicklook section selects			
CQP6	4700	2^0 through 2^{11}	Select sections 00 through 11.
		2^{12} through 2^{15}	Not used.

When Quicklook is selected, sections 0 through 11 are run.

OIT3/P

The following parameters are used by the OIT3/P identification test.

PARAM5	Section Selects - Sections 15 through 00.
PARAM6	Section Selects - Sections 24 through 16.
PARAM7	Section Selects - Not used.
PARAM8	Section Selects - Not used.
PARAM9 through PARAM15	Not used.

When Quicklook is selected (PARAM0 bit 2^{11}), sections 00 through 24 will be run.

PDT3/P

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{15}	Select sections 0 through 5. Default is 77_8.
PARAM6 through PARAM8	130 through 132	2^0 through 2^{15}	Not used.
PARAM16	142	2^0	Bypass BASICTC setting of clock width margins. Default is 0.
		2^1 through 2^{15}	Not used.

When Quicklook is selected, sections 0 through 5 will be run.

SMT3/P

Tag	Octal Address	Bit(s)	Meaning
PARAM5	127	2^0 through 2^{15}	Select sections 0 through 7. Default is 377_8.
PARAM6 through PARAM8	130 through 132	2^0 through 2^{15}	Not used.
PARAM16	142	2^0	Bypass BASICTC setting of clock width margins. Default is 0.
		2^1 through 2^{15}	Not used.

When Quicklook is selected, sections 0 through 7 will be run.

Control Words

Control words are used to identify a program and supply information to the system, to the basic test control code and to diagnostic executive (DEX). Some of the control words must be defined by the individual identification tests at assembly time. The affected control words are described below.

Tag	Octal Address	Meaning
CW0	102	First two characters of the program name in ASCII. Must be defined at assembly time.
CW1	103	Last two characters of the program name in ASCII. Must be defined at assembly time.
CW2	104	Program type: each bit has a meaning. Must be defined at assembly time. The meaning of the bits, as they will relate to identification tests, is provided below: Bit 2^1 Test activates or deactivates RAM features. Bit 2^2 Test uses overlays; should be set for all identification tests. Bits 2^4 through 2^7 form a code. 0 Requires product set microcode. 1 Requires diagnostic microcode. 2 No microcode required. $3\text{-}17_8$ Not used. For the identification tests, most likely value is 1 or 2. Bits 2^8 through 2^{11}: Form a code. Code value for CPU identification tests is 3. Bits 2^{12} through 2^{15}: Form a code. Code of 2 is required for all identification tests and indicates that the program is a detection test.
CW3	105	Current iteration count initialized to zero at beginning of test and at end of subsection.
CW4	106	Error code number 1. The test will store the proper code in CW4 at the beginning of each section, and store new codes whenever appropriate. Refer to error codes for definition.
CW5	107	Error code number 2. Refer to error codes for definition.

Tag	Octal Address	Meaning
CW6	110	Test pass count.
CW7	111	Current section number.
CW8	112	Current subsection number.
CW9	113	Current condition number.
CW10	114	Error count, incremented by one for each error processed by DEX.
CW11	115	Reserved for use by BASICTC.
	.	.
	.	.
	.	.
CW15	121	

Quicklook Section Selection

Each test defines the Quicklook section selects that apply to the individual test. When the test is run in Quicklook mode, the Quicklook section selects bits will be substituted for the section select bits in PARAM5 through PARAM8. Quicklook selects are defined at assembly time.

Tag	Octal Address	Meaning
CQP6	Test defined locations	Select sections 00 through 15.
CQP7	Test defined locations	Select sections 16 through 31.
CQP8	Test defined locations	Select sections 32 through 47.
CQP9	Test defined locations	Select sections 48 through 63.

Test Error Buffer

The test error buffer contains the condition results from the failing subsection. It is the information that is used for error isolation, be it manual or software isolation. The expected contents of each condition result in the error buffer is defined for each error code. Refer to Error Codes.

Tag	Octal Address	Meaning
EBUFNAME	145 and 146	Two locations that contain the test name in ASCII. The first location contains the first two characters of the test name. The second location contains the last two characters of the test name.
EBUFSSS	147	Bits 2^8 through 2^{15} contain the failing section number in hexadecimal. Bits 2^0 through 2^7 contain the failing subsection number in hexadecimal. Limit of 256 sections and 256 subsections/section.

Tag	Octal Address	Meaning
EBUFBITS	150	Condition check failure bits. Each bit corresponds to a condition result within the failing subsection; bit 2^0 is for condition 0 result, bit 2^1 is for condition 1, and so on. When the condition result is not as expected by the test, the corresponding condition check failure bit will be set.
EBUF00	151 through 154	Condition 0 result.
EBUF01	155 through 160	Condition 1 result.
EBUF02	161 through 164	Condition 2 result.
EBUF03	165 through 170	Condition 3 result.
EBUF04	171 through 174	Condition 4 result.
EBUF05	175 through 200	Condition 5 result.
EBUF06	201 through 204	Condition 6 result.
EBUF07	205 through 210	Condition 7 result.
EBUF08	211 through 214	Condition 8 result.
EBUF09	215 through 220	Condition 9 result.
EBUF10	221 through 224	Condition 10 result.
EBUF11	225 through 230	Condition 11 result.
EBUF12	231 through 234	Condition 12 result.
EBUF13	235 through 240	Condition 13 result.
EBUF14	241 through 244	Condition 14 result.
EBUF15	245 through 250	Condition 15 result.

Each condition result consists of four 16-bit PP locations. The distribution of the 64-bit result over the four 16-bit condition result words is as follows:

location X	Bits 00 through 15
location X+1	Bits 16 through 31
location X+2	Bits 32 through 47
location X+3	Bits 48 through 63

Expected Result Buffer

The expected result buffer contains the expected results for each condition executed in a subsection. The expected result buffer information is displayed along with the test error buffer condition results, whenever a failure is detected. Each expected result consists of 64 bits distributed the same as in the error buffer.

Tag	Octal Address	Meaning
EXPECT00	251 through 254	Condition 0 expected results data.
EXPECT01	255 through 260	Condition 1 expected results data.
EXPECT02	261 through 264	Condition 2 expected results data.
EXPECT03	265 through 270	Condition 3 expected results data.
EXPECT04	271 through 274	Condition 4 expected results data.
EXPECT05	275 through 300	Condition 5 expected results data.
EXPECT06	301 through 304	Condition 6 expected results data.
EXPECT07	305 through 310	Condition 7 expected results data.
EXPECT08	311 through 314	Condition 8 expected results data.
EXPECT09	315 through 320	Condition 9 expected results data.
EXPECT10	321 through 324	Condition 10 expected results data.
EXPECT11	325 through 330	Condition 11 expected results data.
EXPECT12	331 through 334	Condition 12 expected results data.
EXPECT13	335 through 340	Condition 13 expected results data.
EXPECT14	341 through 344	Condition 14 expected results data.
EXPECT15	345 through 350	Condition 15 expected results data.

Section Tests

ACT3/P - Address Control Test

Section 0	Verifies primary address path.	
	Subsection 0	Sends zeros and ones to the Untranslatable Pointer Register.
Section 1	Verifies primary address path.	
	Subsection 0	Tests the A/B Operand (ADDRU) data path.
Section 2	Verifies primary address path.	
	Subsection 0	Tests the Address Control Address Offset (OREG) data path and the Address Offset Select Mux (ADDR2).
Section 3	Tests addressing to Soft Control 1 memory (OUTDT1).	
	Subsection 0	Tests Soft Control 1 (OUTDT1) external address path.
	Subsection 1	Tests Soft Control 1 (OUTDT1) internal address path.
Section 4	Does nothing. Kept as a spare.	
Section 5	Tests the 32-bit Address Adder.	
	Subsection 0	Tests intra-array logic.
Section 6	Tests the 32-bit Address Adder.	
	Subsection 0	Tests signals between adjacent arrays.
Section 7	Tests the 32-bit Address Adder.	
	Subsection 0	Tests remaining inter-array signals.
Section 8	Tests the 32-bit Address Adder.	
	Subsection 0	Tests the Address Adder when in C170 mode.
Section 9	Tests the Load/Store Multiple Length (LSMPRE) Adder.	
	Subsection 0	Tests LSMPRE input to the Address Offset Select Mux.
	Subsection 1	Tests LSMPRE Adder, using valid combinations.
	Subsection 2	Tests LSMPRE Adder, using invalid combinations.
Section 10	Tests the LM Byte Address Select Mux (ACAD).	
	Subsection 0	Tests the Mux input from the address adder.
	Subsection 1	Tests the Mux input from the Address Control Stream address counter.

Section 11	Tests error detection logic in the address calculation path.	
	Subsection 0	Tests if parity errors can be forced and detected.
	Subsection 1	Tests parity error detection logic.
Section 12	Tests address specification error and Address Control micrand parity.	
	Subsection 0	Tests the address specification error sensing logic.
	Subsection 1	Tests the Address Control Micrand Register parity bits.
	Subsection 2	Tests the parity detection logic in the Address Control Micrand Register.
Section 13	Tests C170 Mode Address Out of Range (AOR).	
	Subsection 0	Verifies no AOR error occurs when the test address is equal to RAC but less than the RAC+FLC value.
	Subsection 1	Verifies AOR errors are generated when the test address is equal to or greater than FLC.
	Subsection 2	Verifies AOR errors are generated when the test address is less than RAC.
	Subsection 3	Verifies no AOR errors are generated when test address exceeds 18 bits before 8 times RAC is added to it.
Section 14	Tests soft control addressing to the Soft Control 2 Memory (OUTDT2).	
	Subsection 0	Tests OUTDT2 internal address entry.
	Subsection 1	Tests OUTDT2 external address entry.
Section 15	Tests the A Stream data path.	
	Subsection 0	Tests the A Stream data path.
	Subsection 1	Tests the Right Byte Shifter data.
Section 16	Tests byte positioning and selection within the A Stream data path.	
	Subsection 0	Tests the Right Byte Shifter shift count.
Section 17	Tests mark bits generation logic.	
	Subsection 0	Tests mark lines and Address Adder Holding Register.
	Subsection 1	Tests Bit Right Shift Count register and the mark line parity logic.
Section 18	Tests Final Address Control Stream (FAADS) byte number bits used to generate mark bits.	
	Subsection 0	Tests the FAADS input to the Mark Mask/Shift Count Mux.

Section 19	Tests the byte load operation.	
	Subsection 0	Tests the A Stream byte load data path.
Section 20	Tests byte instructions with word boundary.	
	Subsection 0	Tests byte load with Word Boundary Condition.
	Subsection 1	Tests byte store with Word Boundary Condition.
Section 21	Tests Address Control Stream Address Counter and byte operations when run in sequence.	
	Subsection 0	Tests the Address Control Stream address counter.
	Subsection 1	Tests groups of byte instructions.
Section 22	Tests Bit Load data path and control logic.	
	Subsection 0	Tests the Right Bit Shifter and the Final Load Bit entry to the A1/C1 Stream Buffer Register.
Section 23	Tests Bit Store data path and control logic.	
	Subsection 0	Tests the Right Bit Shifter end around circuitry and the Bit Store path.
Section 24	Verifies the Tests and Set Bit instruction.	
	Subsection 0	Tests the Test and Set Bit operation.
Section 25	Tests bit instruction sequences.	
	Subsection 0	Tests combinations of bit operations.
Section 26	Tests parity logic in the A Stream data path.	
	Subsection 0	Ensures that parity bits function and errors can be detected.
	Subsection 1	Tests the parity checkers and generators.

ANT3/P - Arithmetic and Logic Unit (ALN) Test

Section 0	Checks basic Shift Mux and Shift Network.	
	Subsection 0	Tests Shift Mux response to Mux Code E.
	Subsection 1	Tests Shift Network Rank 1 for shift counts of 1, 2, and 4 to bits 57 through 126.
	Subsection 2	Tests Shift Network Rank 1 for shift counts of 1 through 7 to bits 57 through 126.
	Subsection 3	Tests Shift Network Rank 2 for shift counts of 8, 16, and 32 for Rank 2 input bits 32 through 119.
	Subsection 4	Tests Shift Network Rank 2 for shift counts of 8, 16, 24, 32, 40, 48, and 56 for Rank 2 input bits 8 through 119.
	Subsection 5	Tests Shift Network Rank 1 for shift counts of 1 through 7 to bits 1 through 70.
Section 1	Tests Large Adder and does exhaustive test of shift network.	
	Subsection 0	Tests data path from the shifter through the adder to the output Mux, forcing adder Mux to zero, and tests adder OR and ADD functions using a shift of 0 in the shifter.
	Subsection 1	Tests shift count of 8 for the complete shifter using adder and shifter output paths.
	Subsection 2	Tests shift count of 16 for the complete shifter using adder and shifter output paths.
	Subsection 3	Tests shift count of 24 and sign extension for complete shifter using adder and shifter output paths.
	Subsection 4	Tests shift count of 32 and sign extension for complete shifter using adder and shifter output paths.
	Subsection 5	Tests shift count of 40 and sign extension for complete shifter using adder and shifter output.
	Subsection 6	Tests shift count of 48 and sign extension for complete shifter using adder and shifter outputs.
	Subsection 7	Tests shift count of 56 and sign extension for complete shifter using adder and shifter outputs.
	Subsection 8	Tests shift count of 1 for the complete shifter using adder and shifter output paths.
	Subsection 9	Tests shift count of 2 and sign extension for complete shifter using adder and shifter output paths.
	Subsection 10	Tests shift count of 3 and sign extension for complete shifter using adder and shifter output paths.

	Subsection 11	Tests shift count of 4 and sign extension for complete shifter using adder and shifter output.
	Subsection 12	Tests shift count of 5 and sign extension for complete shifter using adder and shifter outputs.
	Subsection 13	Tests shift count of 6 and sign extension for complete shifter using adder and ahifter outputs.
	Subsection 14	Tests shift count of 7, sign extension, and parity prediction using shifter and adder outputs.
Section 2	Checks Shift Mux and Adder Mux.	
	Subsection 0	Tests response of the Shift Mux and Adder Mux to complement code of complement RDSC in Shift Mux.
	Subsection 1	Tests data path of RDSB through Exponent/Integer Mux and Adder Mux and Mux Code 0 control of the Adder Mux and Shift Mux.
	Subsection 2	Tests Shift Mux and Adder Mux response to Mux Code 5.
Section 3	Large Adder boolean and parity checks.	
	Subsection 0	Tests boolean functions in Large Adder bits 32 through 95.
	Subsection 1	Tests boolean function control in Large Adder bits 32 through 95.
	Subsection 2	Does final test of parity logic in Large Adder.
Section 4	Checks Adder Mux and Shift Mux.	
	Subsection 0	Tests Shift Mux response to Mux Code 1.
	Subsection 1	Tests Adder Mux response to Mux Code 1.
	Subsection 2	Tests Extend Zeros to Shift Network and Complement Code 1.
	Subsection 3	Tests Shift Mux and Adder Mux response to Mux Code 2.
	Subsection 4	Tests Shift Mux and Adder Mux response to Mux Code 3.
	Subsection 5	Tests Shift Mux and Adder Mux response to Mux Code 4.
	Subsection 6	Tests Shift Mux and Adder Mux response to Mux Code C.
	Subsection 7	Tests Shift Mux and Adder Mux response to Mux Code D.

	Subsection 8	Tests Shift Mux and Adder Mux response to Mux Code 6.
	Subsection 9	Tests Shift Mux and Adder Mux response to Mux Code 7.
	Subsection 10	Tests Shift Mux and Adder Mux response to Mux Code F.
Section 5	Checks Large Adder and Carry Interchange.	
	Subsection 0	Tests Large Adder and Carry Interchange for proper handling of generates, enables, and satisfies.
	Subsection 1	Tests Carry Interchange for handling of enables.
	Subsection 2	Tests Carry Interchange for handling generates and satisfies End Around.
	Subsection 3	Does final test of Carry Interchange and End Around Carry.
Section 6	Checks Carry Code function for the Carry Interchange and Large Adder.	
	Subsection 0	Tests 18-bit Add Carry Code with sign extension.
	Subsection 1	Tests 60-bit Add Carry Code with sign extension.
	Subsection 2	Tests 32- and 64-bit twos Complement Adds, with automatic plus 1, Carry Codes.
	Subsection 3	Tests 32- and 64-bit twos Complement Adds, with conditional plus 1, Carry Codes.
Section 7	Checks overflow detection and reporting.	
	Subsection 0	Tests 64-bit overflow detection and reporting.
	Subsection 1	Tests 32-bit overflow detection and reporting.
	Subsection 2	Tests 97-bit overflow detection and reporting.
Section 8	Checks Shift Count Generator.	
	Subsection 0	Tests Shift Count Generator Code 6 control.
	Subsection 1	Tests Shift Count Generator Codes 0, 1, and 2 control.
	Subsection 2	Tests Shift Count Generator Codes 3, 4, and 5 control.
Section 9	Checks Shift Count Generator outputs using micrand shift count inputs.	
	Subsection 0	Tests Shift Count Generator for shifts of 48, 32, and 16.
	Subsection 1	Tests Shift Count Generator for shifts of 8 and 4.
	Subsection 2	Tests Shift Count Generator for shifts of 2 and 1.

Section 10		Checks latched shift count from Address Control and Shift Count Generator functions.
	Subsection 0	Tests Address Control shift count data path through Address Control, and Shift Count Generator Codes 6 and 7.
	Subsection 1	Tests Shift Count Generator Codes 0 and 2 with Address Control shift count.
	Subsection 2	Tests Shift Count Generator Codes 1 and 3 with Address Control shift count.
	Subsection 3	Tests Shift Count Generator Codes 4 and 5 with Address Control shift count.
	Subsection 4	Tests Address Control Select of Address Control shift count bit 0.
Section 11		Checks Right Shift 64 generation in Shift Count Generator and its use by the Shift Mux.
	Subsection 0	Tests Right Shift 64 generation and Mux Codes 4 and 5.
	Subsection 1	Tests proper Right Shift 64 generation using Micrand Constant and Address Control shift count.
Section 12		Checks shift fault logic.
	Subsection 0	Tests that Enable Shift Fault does not cause wrong select in output Mux.
	Subsection 1	Tests shift fault detection and shifter output select on shift fault.
Section 13		Checks Shift Count Generator using B and C operand paths.
	Subsection 0	Tests operation of Shift Count Generator Code B.
	Subsection 1	Tests operation of Shift Count Generator Code 9.
	Subsection 2	Tests operation of Shift Count Generator Code 8.
	Subsection 3	Tests operation of Shift Count Generator Code A.
Section 14		Does final check of Shift Count Generator output codes, and Mux code operations that use them.
	Subsection 0	Tests C greater than B and Mux Code 8 operation.
	Subsection 1	Tests Mux Code A operation.
	Subsection 2	Tests Mux Code 6 response to Right Shift 64.
	Subsection 3	Tests Mux Code 9 and SKG Code F.
	Subsection 4	Tests SKG Code E and Adder Result Bit 47 Hold FF.

Section 15	Checks Round Control in Adder Mux and Shift Mux.	
	Subsection 0	Tests Round operation in Shift Mux.
	Subsection 1	Tests Round operation in Adder Mux.
Section 16	Checks Complement Codes 1 and 3.	
	Subsection 0	Tests Complement Code 1.
	Subsection 1	Tests Complement Code 3 in Shift Mux.
	Subsection 2	Tests Complement Code 3 in Adder Mux.
Section 17	Checks the operation of Complement Codes 4, 5, 6, and 7.	
	Subsection 0	Tests Complement Code 4.
	Subsection 1	Tests Complement Code 5.
	Subsection 2	Tests Complement Code 6.
	Subsection 3	Tests Complement Code 7.
Section 18	Checks Normalize Encoder and associated logic.	
	Subsection 0	Tests Normalize Encoder, Mux Code B, and more output data paths and controls.
	Subsection 1	Tests XOR operation in Large Adder bits 0 through 31.
	Subsection 2	Tests Significance Count Adjust Adder.
	Subsection 3	Tests Normalize Control Code 4.
	Subsection 4	Tests operation of micrand bits 24 and 25 and Adder Result Sign.
	Subsection 5	Tests Adder Result Sign Selection in Carry Interchange.
Section 19	Checks the Exponent Arithmetic Network and associated logic.	
	Subsection 0	Tests Exponent Arithmetic Code 2 and Mux Code B.
	Subsection 1	Tests Exponent Arithmetic Code 0.
	Subsection 2	Tests Exponent Arithmetic Code 9 and A.
	Subsection 3	Tests Exponent Arithmetic Code 7 and 8.
	Subsection 4	Tests Exponent Arithmetic Codes 1, 3, and C.
	Subsection 5	Tests Exponent Arithmetic Codes 5 and 6.
	Subsection 6	Tests ability of normalize code to modify exponent arithmetic codes and the Exponent Auxiliary Board to give arithmetic loss of significance.

Section 20	Checks Force Zero out of ALN, nonstandard exponent detection and reporting, and force of nonstandard fixed values out of ALN.	
	Subsection 0	Tests force zeros in the Output Mux.
	Subsection 1	Tests detection and reporting of 180 Exponent Overflow and 170 Infinite.
	Subsection 2	Tests detection and reporting of 180 Exponent Underflow, 180 Indefinite and 170 Indefinite.
	Subsection 3	Tests User Mask Register control of 180 exponent output.
	Subsection 4	Tests detection of Floating Point loss of significance.
Section 21	Checks Mark to Boolean and Enter Signs logic.	
	Subsection 0	Tests Mark to Boolean logic.
	Subsection 1	Tests Signs Per J logic.
Section 22	Checks compare code generation and reporting.	
	Subsection 0	Tests Integer compare.
	Subsection 1	Tests 18-bit compare.
	Subsection 2	Tests Floating Point compare.
Section 23	Checks endcase exponent detection and reporting, User Mask Register and traps on ALN status.	
	Subsection 0	Tests user masks and traps.
	Subsection 1	Tests 180 Subtract Endcase logic.
	Subsection 2	Tests 180 Add Endcase logic.
	Subsection 3	Tests Endcase Intermediate Cycle logic.
	Subsection 4	Tests 180 Multiply Endcase logic.
Section 24	Checks 170 Floating Point endcase detection, reporting and output.	
	Subsection 0	Tests 170 Floating Point add endcase.
	Subsection 1	Tests 170 Floating Point subtract endcase.
	Subsection 2	Tests 170 Floating Point multiply endcase.
	Subsection 3	Tests Detect 170 Integer multiply.
	Subsection 4	Tests 170 Floating Point divide endcase.

Section 25	Checks 180 Floating Point divide endcase and block of Register File write on divide fault.	
	Subsection 0	Tests 180 Floating Point divide endcase and divide fault logic.
Section 26	Checks 170 Shift fault logic and special control on the Exponent Aux from micrand bits 38, 62, and 63.	
	Subsection 0	Tests 170 nominal shift fault detection logic.
	Subsection 1	Tests micrand bits 62 and 38 use of zero hold logic.
	Subsection 2	Tests Double Precision loss of significance logic and remaining Zero Tests outputs from adder.
	Subsection 3	Tests micrand bit 38 block of force zeros output on detection of nonstandard input.
	Subsection 4	Tests ability of micrand bit 63 to force zero output.
	Subsection 5	Tests Force Copy C when micrand bit 38 is set and 170 Indefinite or Infinite is seen on input.
	Subsection 6	Tests remaining logic for detection of Arithmetic Overflow.
Section 27	Checks branch logic in ALN.	
	Subsection 0	Tests copy of ALN branch flip flop to compare code output.
	Subsection 1	Tests branch on 180, B equal to C, B not equal to C, B greater than C and B greater than or equal to C.
	Subsection 2	Tests branch on 170, B equal to C, B not equal to C, B less than C, B greater than or equal to C.
	Subsection 3	Tests 180 exception branch.
	Subsection 4	Tests 170 endcase branch, in range, not in range, definite, indefinite.
	Subsection 5	Tests 180 branch on segments not equal.
	Subsection 6	Tests block of branch on floating compare and branch when an input operand is indefinite.

Section 28	Checks ALN instruction specification error, parity and carry error detectors and parity generators.

Subsection 0	Tests instruction specification error.
Subsection 1	Tests carry error reporting using PTM bit 17.
Subsection 2	Tests parity error reporting from Large Adder.
Subsection 3	Tests parity error detection and reporting on ALN micrand field.
Subsection 4	Tests parity checkers for ALN micrand field.
Subsection 5	Tests Address Control shift count parity error reporting in ALN and Address Control using PTM bit 2.
Subsection 6	Tests Address Control shift count parity checkers and bits 0 through 5 generation.
Subsection 7	Tests exponent parity generation.

BPT3/P - Business Data Processor (BDP) Tests

Section 0	Tests Instruction Specification Error RAM.	
	Subsection 0	Tests the generating of an Instruction Specification error bit.
	Subsection 1	Tests the addressing of the Instruction Specification error RAM and the Rank 32 BDP Descr Type Field lines.
Section 1	Tests Instruction Specification length error.	
	Subsection 0	Tests for Instruction Specification error for length error with the Descr Length Field lines.
Section 2	Tests Basic data path.	
	Subsection 0	Tests the basic data path (A and B Stream data lines to the BDP converted binary lines) along with BDP Cond Bits 2 and 3.
	Subsection 1	Tests the address lines to the Conversion-to Binary/Decimal (Conv-to-Bin/Dec) RAM.
Section 3	Tests Conv-to-Bin/Dec RAM data holding capabilities.	
	Subsection 0	Tests the addressing of the Conv-to-Bin/Dec RAM with an incrementing pattern.
	Subsection 1	Tests further the addressing of the Conv-to-Bin/Dec RAM with a decrementing pattern.
Section 4	Tests Address Control and BDP length counter.	
	Subsection 0	Tests the A Stream of Address Control for a zero length check at the correct time.
	Subsection 1	Tests the length counter in BDP for up to 9 bytes.
Section 5	Tests Byte decode of A Stream in Address Control.	
	Subsection 0	Tests the A Stream byte decode in Address Control while referencing the register file.
	Subsection 1	Tests the A Stream byte decode in Address Control while referencing central memory.
Section 6	Tests Word boundary and length counter of A Stream in Address Control.	
	Subsection 0	Tests the A Stream length counter in Address Control for bytes within the same CM word.
	Subsection 1	Tests the A Stream of Address Control for crossing of word boundaries.

Section 7 Tests right-to-left sequence in A Stream of Address Control

Subsection 0 Tests the A Stream length counter in Address Control for a right-to-left sequence.

Subsection 1 Tests the A Stream of Address Control for crossing word boundaries in a right-to-left sequence.

Section 8 Tests C Stream of Address Control and the C(5) data bits of BDP.

Subsection 0 Tests the C(5) data bit lines to the C Stream of Address Control.

Subsection 1 Tests the byte decode on the C Stream of Address Control without crossing word boundaries.

Section 9 Tests byte control and length counter in Address Control C Stream.

Subsection 0 Tests the length counter and the C(5) data bit register in Address Control and the AK length-equal-to-zero logic in BDP.

Subsection 1 Tests the length counter and byte control for the C Stream of Address Control in a right-to-left sequence.

Section 10 Tests BDP Buffer RAM length counter.

Subsection 0 Tests the length counter bits in BDP for the BDP Buffer RAM.

Subsection 1 Tests the length counter in the A Stream of Address Control for up to $3F_{16}$ bytes for left-to-right and right-to-left.

Section 11 Tests BDP Buffer RAM data.

Subsection 0 Tests the BDP Buffer RAM for holding data correctly.

Subsection 1 Tests the A Stream of Address Control for crossing of word boundaries with 8 through 16 bytes.

Section 12 Tests BDP Buffer RAM addressing.

Subsection 0 Tests the A Stream of Address Control for crossing word boundaries with 8 through 16 bytes, right to left.

Subsection 1 Tests the addressing of the BDP Buffer RAM.

Subsection 2 Further tests the data holding capabilities of the BDP Buffer RAM.

Section 13	Tests word boundary in C Stream of Address Control.	
	Subsection 0	Tests the C Stream of Address Control further for word boundary conditions.
	Subsection 1	Tests the C Stream of Address Control further for word boundary conditions in a right-to-left sequence.
Section 14	Tests Translate Encode RAM.	
	Subsection 0	Tests the referencing of two consecutive register files by the A Stream of Address Control.
	Subsection 1	Tests the Translate Encode RAM.
Section 15	Tests B Stream data path.	
	Subsection 0	Tests the B Stream data path to the BDP input Stream.
	Subsection 1	Tests the B Stream length counter in Address Control for bytes in the same CM word.
Section 16	Tests B Stream word decode in Address Control.	
	Subsection 0	Tests the B Stream word decode in Address Control while crossing word boundaries.
	Subsection 1	Tests the B Stream word decode in Address Control while crossing word boundaries in a right-to-left sequence.
Section 17	Tests B Stream length counter and A Stream addressing.	
	Subsection 0	Tests the B Stream length counter in Address Control.
	Subsection 1	Tests the decrementing of large addresses in the A Stream of Address Control, right-to-left sequence.
Section 18	Tests Large addresses in B Stream of Address Control.	
	Subsection 0	Tests the incrementing of large addresses in the B Stream of Address Control.
	Subsection 1	Tests the decrementing of large addresses in the B Stream of Address Control, right-to-left sequence.
Section 19	Tests Addressing of the A and B Stream Decode RAMs.	
	Subsection 0	Tests the addressing of the A and B Stream Decode RAMs in the BDP and the passing of the Translated EBCDIC bits from the RAM.

Section 20	Tests Decimal/Binary (Dec/Bin) ALUs and Compares of Equal or Greater.	
	Subsection 0	Tests the compare logic for an equal data function check.
	Subsection 1	Tests the compare logic for a greater than function check.
Section 21	Tests X0 and X1 Result Bits and compare for Less Than.	
	Subsection 0	Tests the compare logic for a Less Than function check.
	Subsection 1	Tests the RF10 (X0) Result Bits for correct transmitting.
	Subsection 2	Tests the RF11 (X1) lines for a decimal compare operation.
Section 22	Tests A and B Stream Exhausted signals plus fill.	
	Subsection 0	Tests the A Stream Exhausted and B Stream Exhausted signals plus fill operation.
Section 23	Tests BDP addition and subtraction logic.	
	Subsection 0	Tests the addition logic in the BDP input stream.
	Subsection 1	Tests the subtract logic in the BDP input stream.
Section 24	Tests Form Combined Sign and ASCII Bias.	
	Subsection 0	Tests the Form Combined Sign logic and the ASCII Bias logic in BDP.
	Subsection 1	Tests the recomplementing logic in the BDP output stream.
Section 25	Tests Slack Digit logic in BDP.	
	Subsection 0	Tests the Slack Digit logic in BDP and the B Data signal.
Section 26	Tests Immediate and Immediate One-shot logic.	
	Subsection 0	Tests the Immediate and Immediate One-shot logic in BDP.
Section 27	Tests Scale Left, Scale Right, and Rounding logic.	
	Subsection 0	Tests the Scale Left logic in BDP.
	Subsection 1	Tests the Scale Right locgic and the Rounding logic in BDP.
	Subsection 2	Checks the detection of Decimal Loss of Significance on a Scale Left operation.

Section 28	Tests RF A RAM and Scan Hit Control.	
	Subsection 0	Tests the RF A RAM for data and addressing capabilities.
	Subsection 1	Tests the absence of a Scan Hit indication during a byte scan.
Section 29	Tests RF B bit decoding and Scan Hit.	
	Subsection 0	Tests that a Scan Hit can be generated.
	Subsection 1	Tests for correct bit decoding in the RF B RAM.
Section 30	Tests RF B data and sequence bits.	
	Subsection 0	Tests the X0 Result bits for all of the sequence bits being transmitted.
	Subsection 1	Tests the RF B RAM for proper data and addressing capabilities.
Section 31	Tests Converter Decimal data path.	
	Subsection 0	Tests the basic Converter Decimal data path in BDP.
	Subsection 1	Tests the lower 2 ALUs in BDP Conv-to-Dec adders.
Section 32	Tests Conv-to-Dec/Bin RAM address lines.	
	Subsection 0	Tests bits 2^8 and 2^9 through the Conv-to-Dec Adders in BDP.
	Subsection 1	Tests the address lines for the Conv-to-Dec/Bin RAM in BDP.
Section 33	Tests Converter Multiply by 256 RAM and Conv-to-Dec Adders in convert logic.	
	Subsection 0	Tests the address lines of the Converter Multiply by 256 RAM (times 256 RAM).
	Subsection 1	Tests bits 2^8 to 2^{17} through the BDP convert logic.
Section 34	Tests Converter Results RAM and Carry in Converter Decimal Adders.	
	Subsection 0	Tests the carry operation in the converter logic.
	Subsection 1	Tests the ability of the Converter Results RAM in the converter logic to hold data and be addressed correctly.
	Subsection 2	Tests the moving of all data types from A Stream to all data types on C Stream, and check DLOS status.
	Subsection 3	Tests DLOS status generated with various data types and length counts.
	Subsection 4	Checks the A and B Stream Active signals in BDP.

Section 35 Tests Pass data with the edit logic.

Subsection 0	Tests the passing of data through BDP with an edit operation.
Subsection 1	Tests MOPO for translating data correctly in BDP.

Section 36 Tests Edit Mask and the Symbol Mask RAM.

Subsection 0	Tests the movement of data from the Edit Mask to the C Stream of Address Control.
Subsection 1	Tests the loading and addressing of the Symbol Mask (SM) RAM.

Section 37 Tests Special Character Table RAM in edit logic.

Subsection 0	Tests the loading and holding of data in the Special Character Table (SCT) RAM of BDP.
Subsection 1	Tests the addressing of the SCT RAM.
Subsection 2	Tests edit invalid data status detection.

Section 38 Tests Generating and reporting of BDP parity errors.

Subsection 0	Tests the Scale Counter parity detection.
Subsection 1	Tests the RF A Data parity detection + PTM(3).
Subsection 2	Tests the edit parity detection + PTM(4).
Subsection 3	Tests the sign digit and EBCDIC data parity detection.
Subsection 4	Tests the converter logic parity detection.
Subsection 5	Tests the altering of the Edit Mask by PTM(5).
Subsection 6	Tests the BDP micrand parity detection.
Subsection 7	Tests the Address Control B Stream data parity detection.
Subsection 8	Tests the A and B Stream descriptor lengths parity detection.
Subsection 9	Tests the BDP Convert-to-Binary lines to ALN unit parity detection.
Subsection 10	Tests the SVA latches in Address Control parity detection.

Section 39 Tests Page table hit/miss on virtual BDP operation.

Subsection 0	Tests proper latching in UTP register on hit/miss.

CMT3/P - Memory Test

Section 0	Checks MCH/MAC communication.	
	Subsection 0	Tests basic maintenance channel (MCH) response to function.
	Subsection 1	Tests MCH echo function.
	Subsection 2	Tests MCH data path from IOU and back using echo function.
	Subsection 3	Tests MAC control logic related to Control Word flags.
	Subsection 4	Tests MAC control logic related to Control Bytes Loaded flag.
	Subsection 5	Tests response of MAC logic to a clear errors function.
Section 1	Checks MAC writing and reading capability.	
	Subsection 0	Tests write of all directly accessible processor registers.
	Subsection 1	Tests read of all processor registers.
	Subsection 2	Tests write of all processor Soft Control memories.
	Subsection 3	Tests read of all processor Soft Control memories.
	Subsection 4	Tests write of all processor Register File locations.
	Subsection 5	Tests read of all processor Register File locations.
	Subsection 6	Tests write of all processor Control Store memory.
	Subsection 7	Tests read of all processor Control Store memory.
	Subsection 8	Tests clear of MAC write FF.
	Subsection 9	Tests clear of MAC read FF.
	Subsections 10 through 15	Tests clock.
Section 2	Tests MAC bus data path.	
	Subsection 0	Tests the MAC bus data path from the PFS Register to MCH using zero data.
	Subsection 1	Tests the MAC bus data path within the MAC.
	Subsection 2	Tests the MAC bus data path from Address Control Soft Control 2 through the ALN Soft Control to the IF Decode RAM.
	Subsection 3	Tests the MAC bus data path legs from Register File to BDP Port A and B and from Address Control Soft Control 1 to the BDP RAM 0 and RAM 1.

Section 3	Tests PFS Register.	
	Subsection 0	Tests the byte and board select signals sent to and used on the PFS Register, as well as related control logic on the MAC.
	Subsection 1	Tests the data path from the MAC through the memory maintenance register logic back to the MAC.
	Subsection 2	Tests the gating of data from the memory maintenance registers to the MAC.
	Subsection 3	Tests the multiplexing of upper and lower register selections.
	Subsection 4	Tests the memory maintenance register byte parity generators.
	Subsection 5	Tests memory maintenance register selection.
	Subsection 6	Reads and reports on the Options Installed Register.
Section 4	Tests the memory functions and basic data handling.	
	Subsection 0	Tests basic write/read sequence of memory functions.
	Subsection 1	Tests port input buffer memories.
	Subsection 2	Tests Read and Set Lock function; holds function data to zero.
Section 5	Continues Test of basic memory functions.	
	Subsection 0	Tests Read and Set Lock function; holds memory data to zero.
	Subsection 1	Tests Read and Clear Lock function; holds function data equal F---F$_{16}$.
Section 6	Continues Tests of basic memory functions.	
	Subsection 0	Tests Read and Clear Lock function; holds memory data equal F---F$_{16}$.
Section 7	Tests the SECDED logic.	
	Subsection 0	Tests the ECC generator and read syndrome generator.
	Subsection 1	Tests the partial write ECC generator.
Section 8	Continues test of SECDED Logic.	
	Subsection 0	Tests the SECDED error reporting and correction logic.
Section 9	Tests the detection and logging of errors; forces and verifies errors.	
	Subsection 0	Tests the logging of the syndrome register A0 (CEL).
	Subsection 1	Tests the logging of address bits to register A0.

	Subsection 2	Tests the setting of the unlogged bit in register A0.
Section 10	Continues test of error logs.	
	Subsection 0	Tests the logging of address bits to maintenance register A4 (UNCEL1).
	Subsection 1	Tests the logging of a bounds fault into maintenance register A4.
	Subsection 2	Tests the logging of data out path parity errors in maintenance register A4.
Section 11	Continues test of error logs.	
	Subsection 0	Tests the detection and logging of function parity errors into maintenance register A4.
	Subsection 1	Tests the detection and logging of port parity errors in maintenance register A4.
	Subsection 2	Tests the detection and logging of port parity errors in the port.
	Subsection 3	Verifies that data written with a parity error will be read back without parity error.
	Subsection 4	Tests the partial write logic using CM data having single bit errors.
	Subsection 5	Tests the partial write logic using CM data having multiple bit errors.
	Subsection 6	Tests the memory Status Summary register (SSUM).
Section 12	Continues test of error logs.	
	Subsection 0	Tests the detection and logging of Tag PE into into maintenance register A8 (UNCEL2).
	Subsection 1	Tests the detection and logging of read parity errors into maintenance registers A4 and A8.
Section 13	Tests the bounds detect logic.	
	Subsection 0	Tests address out-of-bounds detection for lower bounds. The test address is greater than the bounds address.
	Subsection 1	Tests address out-of-bounds detection for lower bounds. The test address is equal to one half the bounds.
	Subsection 2	Tests bounds detection for lower bounds. The test address equal to the bounds address.
	Subsection 3	Tests the Lower Bounds Register using addresses equal to twice the bounds address.

Section 14	Continues test of bounds detect logic.	
	Subsection 0	Tests bounds detection for upper bounds. Ensures that a write cannot be performed out-of-bounds. Ensures that a read is not so encumbered.
	Subsection 1	Tests address out-of-bounds for upper bounds. The test addresses will be greater than the bounds address.
Section 15	Continues test of bounds detect logic.	
	Subsection 0	Tests address out-of-bounds for upper bounds. The test addresses will be equal to the upper bounds address.
	Subsection 1	Tests the bounds vector bit (enable bounds).
Section 16	Tests the memory arrays and their support logic.	
	Subsection 0	Tests the data fan-in and Address Control of the fan-in using zero data. Single bit errors are ignored unless they appear to be address related.
	Subsection 1	Tests the data fan-in and Address Control of the fan-in using ones data.
Section 17	Continues test of memory array support logic.	
	Subsection 0	Tests the data fan-in and Address Control of the fan-in using data equal 5---5_{16}.
	Subsection 1	Tests the data fan-in and Address Control of the fan-in using data equal A---A_{16}.
Section 18	Continues test of memory arrays and supporting logic.	
	Subsection 0	Tests memory addressing; writes unique pattern to selected subset of memory. Each memory array is referenced several times.
	Subsection 1	Reads and verifies memory written in subsection 0.
Section 19	Continues test of memory arrays and supporting logic.	
	Subsection 0	Tests memory addressing; same as section 18, except that data is complemented.
	Subsection 1	Tests memory addressing; same as section 18, except that data is complemented.
Section 20	Tests memory arrays.	
	Subsection 0	Not used.
	Subsection 1	Tests the memory arrays. Data patterns 5---5_{16} and A--A_{16} are used.

Section 21	Continues test of memory arrays.	
	Subsection 0	Tests the memory arrays and chip addressing. An address pattern is used.
Section 22	Tests the memory arrays and chip addressing.	
	Subsection 0	Tests the memory arrays and chip addressing. An address complement pattern is used.
Section 23	Tests the memory using a march pattern.	
	Subsection 0	Tests the memory using a march pattern. The march pattern is generated using a Read and Set Lock function.
Section 24	Consists of general tests.	
	Subsection 0	Verifies that write portion of read, modify, write sequence is aborted if an uncorrected error is detected in the read portion of the sequence.
	Subsection 1	Verifies that bit 39 of maintenance register 20_{16} does in fact inhibit entry to the corrected error log.
	Subsection 2	Verifies that the disable parity check bit of maintenance register 20_{16} does function as specified.
	Subsection 3	Test logic that prevents a write to memory when a bounds fault is detected.
	Subsection 4	Tests the data byte parity checkers in non-SECDED mode.
	Subsection 5	Tests the page zero relocation feature of memory.
Section 25	Tests Port Disable logic.	
	Subsection 0	Verifies that the port can be disabled by setting the proper bit in maintenance register 20_{16}.
	Subsection 1	Verifies that the other ports can be disabled without effecting the IOU port.
Section 26	Utility section.	
	Subsection 0	Not used.
Section 27	Utility section.	
	Subsection 0	Not used.
Section 28	Tests central memory addressing.	
	Subsection 0	Uses a limited number of address bits to verify that all of the address lines work properly. Not all addresses are referenced.

CST3/P - Control Store Basic Paths Test

Section 0	Tests basic Control Store operations.	
	Subsection 0	Tests the start and stop logic during Control Store Sweep.
	Subsection 1	Tests the Control Store Breakpoint Comparator logic.
	Subsection 2	Tests Control Store Step Control logic while sweeping.
	Subsection 3	Tests the MSC Parity Error/Fatal Error logic.
Section 1	Tests basic micrand execution by Control Store.	
	Subsection 0	Performs initial test of online micrand execution.
	Subsection 1	Performs second test of online micrand execution to ensure that Control Store Sweep is not always enabled.
	Subsection 2	Tests MIS=1 translation logic and MIS=1 setting Halt flag.
Section 2	Tests the Control Store branching logic.	
	Subsection 0	Performs first test of the Control Store Branch Delta Adder and the adder's Parity Predictor logic.
	Subsection 1	Performs second test of the Control Store Branch Delta Adder and the adder's Parity Predictor logic.
	Subsection 2	Performs final test of the Control Store Branch Delta Adder and the adder's Parity Predictor logic.
	Subsection 3	Tests the branch to BA micrand execution and the BA field data bits path through the Control Store Micrand Address Mux1.
Section 3	Tests the loading and use of the Control Store Return Register.	
	Subsection 0	Tests Control Store's Return Register logic when entering the Return Register with the current micrand address.
	Subsection 1	Tests loading the Return Register with the current micrand address +1.
Section 4	Tests basic operation of copying FU micrand data to the Register File.	
	Subsection 0	Tests the transfer of FU micrand data to the Register File.
	Subsection 1	Tests the RDSa to RDSw OPI Minipipe and the OPI Register File addressing when writing Register File with Immediate FU micrand data.
	Subsection 2	Tests that no valids are enabled into the Control Pipe when Control Store is offline, and also when MIS field is 1 (Block Valid and set Halt flag).

Section 5 Tests basic operation of transferring data from the Register File, through the ALN unit and back to the Register File.

- Subsection 0 Tests the basic Register File select paths for C operand and B operand to ALN, through ALN, back to DAI Mux Register and to Register File.
- Subsection 1 Tests the OPI RDS Field Select Mux (RDSc leg) and the Register File Address Select Mux (literal address leg).
- Subsection 2 Tests the OPI RDS Field Select Mux (RDSb leg).

Section 6 Tests the basic registers in the Segment Map Unit.

- Subsection 0 Tests the data path from RDSa operand select to Segment Map's PVA Register, New P Register, Old P Register and back to OPI Live Register Read Mux, then to the OPI Operand Data Mux.
- Subsection 1 Tests the parity checkers along the data path from RDSa operand select to Segment Map's PVA Register, New P Register, Old P Register and back to OPI Live Register Read Mux, then to the OPI Operand Data Mux, through ALN, DAI Mux/Register and to the Register File.
- Subsection 2 Tests the parity error reporting from the parity checkers for Segment Map's PVA Register, New P Register, Seg Descriptor Mux bits 16 through 31 and Address Control A/C Stream ASID Register.

Section 7 Tests basic processor read and write central memory operations.

- Subsection 0 Tests the RDSa data path from OPI to Address Control Recovery Address Register, the Untranslatable Pointer Register (bits 32 through 63), the OPI Live Register Read Mux, and finally OPI Operand Data Mux.
- Subsection 1 Tests the basic read data path from central memory to the Register File.
- Subsection 2 Tests the processor read memory address path to central memory.
- Subsection 3 Tests the RDSd select bits path through the OPI RDS Write Selector Mux (RDSw) and partial test of RDSa literal data through OPI Immediate Operand Mux.
- Subsection 4 Tests the word write data path from OPI C Operand Register to CMC/CM.

Section 8 Tests processor logic involved in segment descriptor fetching.

- Subsection 0 Tests the Segment Map Segment Descriptor Input Register, Segment Map Segment Descriptor Mux Shortstop leg, and the Segment Map Segment Descriptor Mux parity checkers.

	Subsection 1	Tests that the Segment Map Seg Descriptor Mux parity checkers report parity errors.
	Subsection 2	Perform a basic test of the Segment Descriptor Fetch logic.
	Subsection 3	Tests the Segment Map P Descriptor Save Register data holding capability.
Section 9	Tests Control Store conditional branching.	
	Subsection 0	Tests the Segment Map C170 Mode flag and UVMID flag as well as Control Store C170 Mode and UVMID conditional branching.
	Subsection 1	Tests Control Store Segment Map P Descriptor Local and/or Global Privilege conditional branching.
	Subsection 2	Tests Control Store conditional branching when P Ring Number Initial is equal to P Ring Number Final, and the Segment Map Old/New P Ring Comparator.
	Subsection 3	Tests Control Store Conditional Branching when Halt flag is set and when Halt flag is clear.
	Subsection 4	Tests the toggling of the C180 Monitor Mode flag and Control Store conditional branching on C180 Monitor Mode Condition.
	Subsection 5	Tests the DAI Condition Select Register and Control Store conditional branching for different DAI Condition Select Register bit settings.
	Subsection 6	Performs final test of Control Store conditional branching when Tests Mode (DEC bit 33) is set and clear.
	Subsection 7	Initializes subsection 9 conditions.
	Subsection 8	Tests Control Store conditional branching when checking forced condition false.
	Subsection 9	Tests that Control Store conditional branching to BD equals 0 and condition selects of 0C through 0F and 1C through 1F will block the pipe valid signal.
	Subsection 10	Completes testing of the Control Store Special Condition Select (SBDEQZ) logic circuit.
Section 10	Tests basic processor instruction fetch operation.	
	Subsection 0	Tests Instruction Fetch Initialization and Control Store R125 Invalid Trap logic.
	Subsection 1	Tests that Control Store Pipe Valid is blocked when performing exit with R125 not valid.
	Subsection 2	Performs initial test of the Control Store Exit micrand address formation for C180 and C170 Modes.

	Subsection 3	Tests Instruction Fetch (IF) P Register to ICP P Registers to OPI Live Register Read Mux data paths for P value of 0.
	Subsection 4	Tests the writing of IF P Register from the DAI Register and completes the testing of the data path from IF P Register through the Pipe to OPI Live Register Read Mux.
	Subsection 5	Tests the instruction word data path from central memory to IF.
	Subsection 6	Tests the IF address data path from IF to local memory.
Section 11	Tests remaining Control Store logic involved with fetching virtual instructions.	
	Subsection 0	Tests the C170 Op Code data path from IF C170 Decode RAMs, through the IF Buffer Word Assembly, and through the Control Store R125 address formation logic to the Control Store Micrand Address Mux1.
	Subsection 1	Tests C170 Op Code to Control Store micrand address formation for C170 store instructions with and without Purge Instruction Stack Flag set.
	Subsection 2	Tests Control Store micrand address formation when executing an exit to an unimplemented C180 instruction.
	Subsection 3	Tests Control Store micrand address formation when executing an exit to a nonexecutable instruction.
	Subsection 4	Does nothing subsection. Can be used for later test expansion.
	Subsection 5	Tests proper operation of Control Store logic when micrand's GME field is equal to 2, void all pipe ranks and initialize instruction fetch.
Section 12	Tests basic microcode trap operation.	
	Subsection 0	Performs initial test of the Monitor Condition Register (MCR) for data holding capability. Also tests writing, reading, and read/clear of MCR.
	Subsection 1	Performs final test of the MCR data holding capability.
	Subsection 2	Performs initial test of the User Condition Register (UCR) for data holding capability. Also tests writing, reading, and read/clear of UCR.
	Subsection 3	Performs final test of the UCR data holding capability.
	Subsection 4	Tests that Write Monitor Condition Mask Register and Write User Condition Mask Register micrands execute without hanging up.

Subsection 5	Tests ICP Dead Man Timer, ICC Monitor Condition Register bit 0, ICC Trap Valid signal and partially tests Control Store Trap logic.
Subsection 6	Completes testing of the Control Store Trap logic and partially tests the ICC Micro Trap Code formation.
Subsection 7	Tests the ICC logic to verify that the Trap Valid signal drops after causing a Micro Trap.

CTT3/P - Common CST/MAC Logic Test

Section 0	Tests stop control.	
	Subsection 0	Tests that Control Store logic operates correctly when a Stop MCH function is received and conditional exit micrands are being executed.
	Subsection 1	Tests that Control Store logic operates correctly when a Stop MCH function is received and unconditional exit micrands are being executed.
Section 1	Tests MAC/Control Store logic involved with microcode assisted read operations.	
	Subsection 0	Tests that a MAC microcode assisted read request to Control Store causes a Micro Trap to CST address 30_{16} while conditional exit micrands are being executed.
	Subsection 1	Tests that a MAC microcode assisted read request to Control Store causes a Micro Trap to CST address 30_{16} while Control Store is spinning on a Set Halt Flag micrand.
	Subsection 2	Tests the MAC reference ROM data path to Control Store Micrand Address Register and the Control Store Branch to CST address indexed by reference ROM micrand control logic.
	Subsection 3	Tests the MAC addressing of the MAC reference ROM RAM for all microcode assisted register numbers.
	Subsection 4	Tests that a complete MAC microcode assisted read operation executes correctly when the processor is executing conditional exits.
	Subsection 5	Tests that a complete MAC microcode assisted read operation executes correctly when the processor is online and halted. Also test MAC lockout logic in Control Store.
	Subsection 6	Tests that a microcode assisted read request will not be honored by Control Store while unconditional exit micrands are being executed.
	Subsection 7	Tests the Control Store data path from MSC Register bits 6 through 10 to bits 0 through 4 of Micrand Address MUX2 Select, Select Reference ROM Address leg.
	Subsection 8	Tests that MAC RF ROM can be used as an entry point to the other half of Control Store (extended or normal), with CAE = 5.

Section 2	Tests MAC/Control Store logic involved with microcode assisted write operations.	
	Subsection 0	Tests MAC and Control Store logic involved with microcode assisted write operation.
	Subsection 1	Completes the testing of MAC and Control Store microcode assisted write logic.
	Subsection 2	Tests that the ICC logic properly handles reporting deadman timeouts which occur during MAC microcode assisted operations.
	Subsection 3	Tests the ICC logic which prevents writing into MCR and UCR when they are nonzero unless the writing is done as a MAC microcode assisted write.
Section 3	Tests MAC/Control Store logic involved with microcode reading and writing MAC resident registers.	
	Subsection 0	Tests Control Store conditional branching logic for MAC Busy condition and the ability of Control Store to set MAC Busy FF.
	Subsection 1	Tests that a MCH read of a directly accessible Processor Register, via the Processor MAC, causes the MAC Busy FF to set and that the MAC Busy FF is cleared when the read operation is complete (8 bytes transferred).
	Subsection 2	Tests the OPI Reference ROM Address Register and the data path from the reference ROM Address Register to MAC reference ROM.
	Subsection 3	Tests the OPI and MAC logic involved in an IU Read MAC Resident Register operation.
	Subsection 4	Completes testing of MAC Data MUX5 to Data MUX6 data path.
	Subsection 5	Tests the OPI and MAC logic involved in an IU Write MAC Resident Register operation.
Section 4	Tests processor debug related logic.	
	Subsection 0	Tests the data holding capabilities of the Debug Mask Live Register and its parity checker.
	Subsection 1	Tests Debug Mask Parity Checker error reporting.
	Subsection 2	Performs initial test of Debug Hit logic.
	Subsection 3	Tests Debug Lockout logic.
	Subsection 4	Tests the Control Store logic that blocks the setting of the exit FF and causes a microtrap to Control Store address $7FF_{16}$ when a Conditional Exit micrand is encountered with the Debug Lockout FF set.

	Subsection 5	Tests that Control Store Debug Lockout FF is cleared by an exit micrand.
	Subsection 6	Tests that a Debug Hit with micrand's MIS field equal to 2 will inhibit the writing of a live register.
	Subsection 7	Tests that Debug Hits occur correctly for different combinations of the Debug Mask Register bits and debug profile bits.
	Subsection 8	Tests MAC microcode assist requests to Control Store when Control Store is performing debug conditional exits.
Section 5	Tests Control Store branch address parity regeneration logic.	
	Subsection 0	Completes testing of the Control Store branch address parity regeneration upper logic.
	Subsection 1	Completes testing of the Control Store branch address parity regeneration lower logic.

ICT3/P - Instruction Complete Control, Instruction Control Pipeline Test

Section 0	Tests Stream Mode logic in OPI.	
	Subsection 0	Performs initial test of Stream Mode logic in OPI.
	Subsection 1	Tests Load Control circuitry for the Process Interval Timer (PIT) and STA live registers in Stream Mode.
	Subsection 2	Tests the Stream Mode Mux and the remaining live register write enable logic for Stream Mode.
Section 1	Performs initial test of interrupt evaluation logic.	
	Subsection 0	Tests that Point-of-No-Return (PONR) can be taken without taking erroneous interrupts.
	Subsection 1	Performs initial test of transfer of condition registers.
	Subsection 2	Tests the Before FF.
	Subsection 3	Tests that no PONR signals are stuck active to enable unwanted interrupts from the MCR or UCR.
Section 2	Tests interrupts from the Rank 60 MCR and UCR.	
	Subsection 0	Tests the MCR Unmasked and Not Trap Interrupt State and MCR With Mask Clear Halt signal generation from the Rank 60 MCR.
	Subsection 1	Tests the generation of the MCR Exchange/Trap/Stack signal from the Rank 60 MCR.
	Subsection 2	Tests generation of the Masked UCR Trap/Stack signal from the Rank 60 UCR.
	Subsection 3	Tests generation of the Unmasked UCR Trap/Exchange/Halt Interrupt signal from the Rank 60 UCR.
	Subsection 4	Tests Halt Interrupts from the Rank 60 MCR and UCR.
	Subsection 5	Tests Exchange Interrupts from the Rank 60 MCR and UCR.
	Subsection 6	Tests Trap Interrupts from the Rank 60 MCR and UCR.
Section 3	Tests the Rank 50 MCR and UCR, Rank 50 and Rank 60 ECR, and interrupts from these registers.	
	Subsection 0	Tests interrupts from Rank 50 MCR bits 0, 3, and 4, and the transfer of those bits to Rank 60.
	Subsection 1	Tests interrupts from Rank 50 MCR bits 6, 7, 9, 12, and 13, and the transfer of those bits to Rank 60.
	Subsection 2	Tests Rank 50 MCR bit 5, and Exchange Request and Accept logic in OPI and in the IOU.

	Subsection 3	Tests interrupts from Rank 50 MCR bit 5, transfer of that bit to Rank 60, 170 Exchange Interrupts, and the Exchange Accept signal to the IOU.
	Subsection 4	Tests External Interrupts from the IOU, and MCR bit 8.
	Subsection 5	Tests interrupts from Rank 50 UCR bits 7, 9, 13, 14, and 15, and the transfer of those bits to Rank 60.
	Subsection 6	Tests interrupts from the Rank 50 and Rank 60 ECR, and transfer from the Rank 50 to the Rank 60 ECR.
Section 4	Tests Retry logic.	
	Subsection 0	Tests that PTM bit 8 can set Rank 50 MCR(0) via Tests Retry.
	Subsection 1	Tests Retry Interrupt generation and Retry Count parity error checking.
	Subsection 2	Tests Retry Count loading and decrement, and determination of Successful Retry.
	Subsection 3	Tests the Shadow Memory Retry in both halves of CS.
	Subsection 4	Tests control transfer to extended Control Store with VMID.
Section 5	Performs final test of interrupt priority logic.	
	Subsection 0	Performs exhaustive test of the interrupt priority logic and microtrap code parity.
Section 6	Tests Exchange Interrupt State and Trap Interrupt State.	
	Subsection 0	Tests the Exchange Interrupt State FF and the Exchange Interrupt FF.
	Subsection 1	Tests remaining untested logic for setting and clearing the Trap Interrupt State FF.
	Subsection 2	Tests setting of Rank 60 MCR bit 15 in Trap Interrupt State.
Section 7	Tests Clear Pipe and Inhibit Write Logic.	
	Subsection 0	Tests Clear Pipe generation, and inhibit of Register File and Live Register writes in OPI.
	Subsection 1	Tests Branch Not Taken signal from ALN through IF to ICC, generation of the Gated Clear Pipe signal, and that Gated Clear Pipe blocks clearing of the condition registers.
	Subsection 2	Tests Gated Clear Pipe fan out to ICP, and the Clear Pipe FF in ICP.

	Subsection 3	Tests generation of LM Write Abort, and its effect in central memory.
Section 8	Tests Unbranch in IF and Rank 50 P Register in ICP.	
	Subsection 0	Tests the P Register bits 32-63 data path through Rank 50, and the Interrupt signal from ICC to IF.
	Subsection 1	Tests the ICP Rank 50 input to the IF Branch Address Adder A Input Mux. The +4 input to the Branch Address Adder B Input Mux is also tested.
Section 9	Performs exhaustive test of operand conflict resolution logic in OPI.	
	Subsection 0	Tests the RDSA and RDSB Recirculate Data Paths.
	Subsection 1	Tests the Register File Shortstop data path and the Rank 40 and Rank 50 comparators in the minipipe.
	Subsection 2	Tests the Operand Shortstop data paths between the ALN, Address Control, and LM functional units, and the B and C Operand Registers.
	Subsection 3	Tests the Rank 33 comparator in the minipipe.
	Subsection 4	Tests the Reload data path.
	Subsection 5	Tests generation of the B Operand Conflict signal and associated logic.
Section 10	Performs final test of the logic which enables faults into the Rank 50 Condition Registers.	
	Subsection 0	Tests the logic which enables the loading of faults into the Rank 50 Condition Registers.
Section 11	Tests operation of LM Write Abort Hold logic and Disable the Escaped Micrand logic.	
	Subsection 0	Tests LM Write Abort Hold and A Stream Abort Enable.
	Subsection 1	Tests Disable the Escaped Micrand operation.
Section 12	Tests decrementing of System Interval Timer (SIT) and PIT registers.	
	Subsection 1	Tests the SIT and PIT for correct decrementing.

IFT3/P - Instruction Fetch Test

Section 0	Tests Basic P path and parity.	
	Subsection 0	Tests the Master Clear processor function.
	Subsection 1	Tests Initialize of IF.
	Subsection 2	Tests entry of New P from OPI.
	Subsection 3	Tests P path parity error reporting.
Section 1	Tests CM load to IF.	
	Subsection 0	Tests initial CM load to IF.
Section 2	Tests Instruction Assembly Mux.	
	Subsection 0	Tests C180 instruction leg of Mux.
	Subsection 1	Tests C170 even RAM input to Mux.
	Subsection 2	Tests C170 odd RAM input to Mux.
	Subsection 3	Tests C170 leg of non-opcode portion of Mux.
Section 3	Tests parcel select multiplexors.	
	Subsection 0	Tests C180 parcel select Mux.
	Subsection 1	Tests C170 parcel select Mux.
Section 4	Tests P register incrementer.	
	Subsection 0	Tests the incrementer with +2.
	Subsection 1	Tests the incrementer with +4.
Section 5	Tests the Branch Control Logic.	
	Subsection 0	Tests branch indexed using Instruction Mux bit 42.
	Subsection 1	Tests branch control with various target addresses.
Section 6	Tests branch address adder input multiplexors.	
	Subsection 0	Instruction BFR RK 2 to Adder.
	Subsection 1	C180 Q times 2 input to adder.
	Subsection 2	C170 reference address input to adder.
	Subsection 3	C170 K times 4 input to adder.
Section 7	Tests branch address adder.	
	Subsection 0	Tests branch address adder.

Section 8	Tests AOR.	
	Subsection 0	Tests the C170 Address-out-of-Range (AOR) detection.
	Subsection 1	Tests AOR by sweeping through all addresses up to the maximum RA + FLC.
Section 9	Performs miscellaneous tests.	
	Subsection 0	Tests gating of the Parcel 3 Save Register.
	Subsection 1	Tests parity checkers following Instruction Buffer ranks 2 and 12.
	Subsection 2	Tests BDP input leg of the Instruction Mux.
	Subsection 3	Tests BDP control using two descriptors.
	Subsection 4	Tests the parcel request control logic.
	Subsection 5	Tests the Instruction Buffer ranks 3, 10, and 11.
	Subsection 6	Tests the Address Specified Error detect logic, C170 mode.
	Subsection 7	Tests the Address Specified Error detect in C180 mode.
	Subsection 8	Tests the IFPURG signal to instruction fetch.

LMT3/P - Local Memory Test

Section 0	Tests RMA address path.	
	Subsection 0	Tests page size mask.
	Subsection 1	Tests stream mode tag bits and CMC tag to CMC path.
	Subsection 2	Tests the CMC Free Running Counter.
	Subsection 3	Tests CMC Refresh and Interrupt functions.
	Subsection 4	Tests RMA Bank Select/CM Busy.
Section 1	Tests parity circuits using RMA addresses.	
	Subsection 0	Tests LMTAG/CMTAG parity.
	Subsection 1	Tests CM fault responses (part 1).
	Subsection 2	Tests CM fault responses (part 2).
	Subsection 3	Tests Page Offset/PSM parity.
	Subsection 4	Tests RMA parity generation logic.
	Subsection 5	Tests RMA parity error signals.
Section 2	Tests preliminary cache checks.	
	Subsection 0	Tests ACADDR/IFADDR through AMUX to Cache Sets 0 and 1.
	Subsection 1	Tests ACADDR/IFADDR through AMUX to Cache Sets 2 and 3.
Section 3	Tests cache data RAMs data integrity.	
	Subsection 0	Tests Set 0 data test.
	Subsection 1	Tests Set 1 data test.
	Subsection 2	Tests Set 2 data test (if available).
	Subsection 3	Tests Set 3 data test (if available).
Section 4	Tests cache tag RAMs data integrity.	
	Subsection 0	Tests Set 0 tag RAMs.
	Subsection 1	Tests Set 1 tag RAMs.
	Subsection 2	Tests Set 2 tag RAMs (if available).
	Subsection 3	Tests Set 3 tag RAMs (if available).

Section 5	Tests cache tag compare network.	
	Subsection 0	Tests Set 0 comparator.
	Subsection 1	Tests Set 1 comparator.
	Subsection 2	Tests Set 2 comparator (if available).
	Subsection 3	Tests Set 3 comparator (if available).
Section 6	Tests cache addressing, external to RAM.	
	Subsection 0	Tests Set 0 external addressing.
	Subsection 1	Tests Set 1 external addressing.
	Subsection 2	Tests Set 2 external addressing (if available).
	Subsection 3	Tests Set 3 external addressing (if available).
Section 7	Tests cache RAM addressing, tag and data.	
	Subsection 0	Tests Set 0 RAM addressing.
	Subsection 1	Tests Set 1 RAM addressing.
	Subsection 2	Tests Set 2 RAM addressing (if available).
	Subsection 3	Tests Set 3 RAM addressing (if available).
Section 8	Tests Purge Cache All.	
	Subsection 0	Tests Set 0 Purge All.
	Subsection 1	Tests Set 1 Purge All.
	Subsection 2	Tests Set 2 Purge All (if available).
	Subsection 3	Tests Set 3 Purge All (if available).
Section 9	Tests Cache Usable signal from MAC.	
	Subsection 0	Tests Set 0 Cache Usable.
	Subsection 1	Tests Set 1 Cache Usable.
	Subsection 2	Tests Set 2 Cache Usable (if available).
	Subsection 3	Tests Set 3 Cache Usable (if available).
Section 10	Tests cache parity circuits.	
	Subsection 0	Tests Set 0 parity circuits.
	Subsection 1	Tests Set 1 parity circuits.
	Subsection 2	Tests Set 2 parity circuits (if available).
	Subsection 3	Tests Set 3 parity circuits (if available).
	Subsection 4	Tests cache write data parity Checker.

Section 11	Tests cache parity error signals.	
	Subsection 0	Tests cache write data parity signals.
	Subsection 1	Tests mark data parity signals.
	Subsection 2	Tests Cache Address parity signals.
	Subsection 3	Tests Cache Address Register (CAR) PE, and Tag-In PE signals.
	Subsection 4	Tests CHC Status verification.
	Subsection 5	Tests Cache multi-hit test.
	Subsection 6	Tests Cache RAM parity signals (CRAMPE).
Section 12	Tests Purge cache functions.	
	Subsection 0	Tests Purge Cache Block (512 bytes).
	Subsection 1	Tests Purge Cache by ASID.
	Subsection 2	Tests Purge Cache by ASID using upper CAR address.
Section 13	Tests Cache Allocate and LRU Network.	
	Subsection 0	Tests Cache Allocate by Set Not Valid.
	Subsection 1	Tests Cache Status RAM (LRU code only) data test.
	Subsection 2	Tests Cache LRU Update network.
	Subsection 3	Tests Status RAM parity signals.
Section 14	Tests Address Control A Stream Counter.	
	Subsection 0	Tests Address Control A Stream Counter.
Section 15	Tests Page Table Address/Length Registers.	
	Subsection 0	Performs PTA/PTL test, Part 1.
	Subsection 1	Performs PTA/PTL test, Part 2.
Section 16	Tests Page Table Address to RMA Mux.	
	Subsection 0	Performs initial PTA to RMA Mux, Part 1.
	Subsection 1	Performs initial PTA to RMA Mux, Part 2.
Section 17	Tests hash code logic network.	
	Subsection 0	Tests the Hash Shifter.
	Subsection 1	Tests the PTA Exclusive OR (EXCL OR) logic.

Section 18	Performs RMA incrementer test.	
	Subsection 0	Searches up to 32 entries.
	Subsection 1	Completes the incrementer test.
Section 19	Tests RMA Mux, Incrementer Logic Merge.	
	Subsection 0	Tests RMA Mux, Adder Merge, Part 1.
	Subsection 1	Tests RMA Mux, Adder Merge, Part 2.
Section 20	Tests PTA/PTL parity circuits.	
	Subsection 0	Tests PTA/PTL parity checkers.
	Subsection 1	Tests PTA/PTL and INCC parity signals.
Section 21	Tests Page Table hit comparator.	
	Subsection 0	Tests Page Table hit comparator, Match Valid.
	Subsection 1	Tests Page Table hit comparator, Match Not Valid.
Section 22	Tests RMA Mux, Page Frame Address entry.	
	Subsection 0	Tests RMA Mux, Page Frame Address entry, Part 1.
	Subsection 1	Tests RMA Mux, Page Frame Address entry, Part 2.
Section 23	Tests MAP PFA/PO merge input to Final Mux Register.	
	Subsection 0	Tests PFA/PO merge to final RMA Mux, Sets 0 and 1.
	Subsection 1	Tests PFA/PO merge to final RMA Mux, Set 2.
	Subsection 2	Tests Page Table hit, Prevalidated.
Section 24	Tests Page Map data integrity.	
	Subsection 0	Tests Page Map data test, Zeros Pattern.
	Subsection 1	Tests Page Map data test, Ones Pattern.
	Subsection 2	Tests Page Map data test, Alternate Bits 1.
	Subsection 3	Tests Page Map data test, Alternate Bits 2.
Section 25	Tests Page Map addressing.	
	Subsection 0	Tests Set 0 Page Map addressing.
	Subsection 1	Tests Set 1 Page Map addressing.
	Subsection 2	Tests Set 2 Page Map addressing.

Section 26	Tests Purge Page Map All.	
	Subsection 0	Tests Set 0 Purge Map All test.
	Subsection 1	Tests Set 1 Purge Map All test.
	Subsection 2	Tests Set 2 Purge Map All test.
Section 27	Completes Map Purge test.	
	Subsection 0	Tests Purge by ASID, Purge by Page.
Section 28	Tests Page Map comparator logic.	
	Subsection 0	Tests Set 0 Page Map comparator.
	Subsection 1	Tests Set 1 Page Map comparator.
	Subsection 2	Tests Set 2 Page Map comparator.
Section 29	Tests Page Map status RAM and Allocate test.	
	Subsection 0	Tests Page Map Allocate and LRU Update.
	Subsection 1	Tests Page Map status RAM data and address test.
Section 30	Tests Page Map parity logic.	
	Subsection 0	Tests Page Map (SPID/PFA) parity circuits.
	Subsection 1	Tests Page Map parity error signals (MPASPE).
	Subsection 2	Tests Page Map parity error signal (MPPFPE).
Section 31	Tests Page Table Search and basic block fill.	
	Subsection 0	Tests Page Table Search, Even/Odd address test.
	Subsection 1	Tests Block read address (WBKAB) test.
Section 32	Tests CM to Cache Data Path and Tag File test.	
	Subsection 0	Tests Tag File 0, II/SA Requests.
	Subsection 1	Tests Tag File 0, II/IF Requests.
	Subsection 2	Tests Tag File 1, II/SA Requests.
	Subsection 3	Tests Tag File 1, II/IF Requests.
	Subsection 4	Tests Tag File 2, II/SA Requests.
	Subsection 5	Tests Tag File 2, II/IF Requests.
	Subsection 6	Tests Tag File 3, II/SA Requests.
	Subsection 7	Tests Tag File 3, II/IF Requests.

Section 33	Tests Cache Hit Valid logic.	
	Subsection 0	Tests Cache Hit Valid, Sets 0 and 1.
	Subsection 1	Tests Cache Hit Valid, Sets 2 and 3.
Section 34	Tests Cache Hit Not Valid and Associated RAMs.	
	Subsection 0	Tests Set 0 Cache Hit Not Valid and RAM test.
	Subsection 1	Tests Set 1 Cache Hit Not Valid and RAM test.
	Subsection 2	Tests Set 2 Cache Hit Not Valid and RAM test.
	Subsection 3	Tests Set 3 Cache Hit Not Valid and RAM test.
Section 35	Tests CMC tag decoder Rank A/Rank B logic and data.	
	Subsection 0	Tests Rank A response network.
	Subsection 1	Tests Rank B logic and data, Part 1.
	Subsection 2	Tests Rank B logic and data, Part 2.
Section 36	Tests FAKECM and fault responses.	
	Subsection 0	Tests FAKECM responses.
	Subsection 1	Tests Page Table fault responses.
	Subsection 2	Tests Page faults with double response and test operations.
	Subsection 3	Tests Special INDEX/TEST situations
	Subsection 4	Tests Unique situation of memory request lockout.
Section 37	Tests CM Invalidate/Cache Bypass.	
	Subsection 0	Tests CM Invalidate and Cache Purge Four-Word Block.
	Subsection 1	Tests Cache Bypass.
	Subsection 2	Tests CM Invalidate bits continued.
Section 38	Tests Tag File, AMUX, Block Fill data parity circuits.	
	Subsection 0	Tests Tag File parity signal verification (TFADPE,TAGFPE).
	Subsection 1	Tests SELRD Status Code and Block Fill data parity signal (CHCMPE) verification.
	Subsection 2	Tests Address Mux parity error code verification (AMUXPE).

Section 39	Tests Cache Lookahead, Part 1.	
	Subsection 0	Tests Cache Lookahead, NXTBA to AMUX.
	Subsection 1	Tests Cache Lookahead, verify operation.
	Subsection 2	Tests Cache Lookahead, NXTBA Incrementer.
Section 40	Tests Cache Lookahead, Part 2.	
	Subsection 0	Tests Set 0, Status RAM bit 2 (BKUSED).
	Subsection 1	Tests Set 1, Status RAM bit 2 (BKUSED).
	Subsection 2	Tests Set 2, Status RAM bit 2 (BKUSED).
	Subsection 3	Tests Set 3, Status RAM bit 2 (BKUSED).
Section 41	Tests Cache Lookahead, Part 3.	
	Subsection 0	Tests Inhibit Page Fault after Lookahead.
	Subsection 1	Tests Unconditional/Conditional Lookahead Select.
	Subsection 2	Tests Memory Requests Outstanding, PTCT Counter.
Section 42	Tests Page Table update.	
	Subsection 0	Tests Update used bit in Page Table.
	Subsection 1	Tests RMA Decrement for Update Address.
	Subsection 2	Tests Update Modify during a Map Miss Write.
	Subsection 3	Tests Update Modify in Map from Page Table.
Section 43	Tests Page Map status RAM, Modity Bit.	
	Subsection 0	Tests Modify Bit integrity, Map Set 0.
	Subsection 1	Tests Modify Bit integrity, Map Set 1.
	Subsection 2	Tests Modify Bit integrity, Map Set 2.
Section 44	Tests Page Map status parity circuits.	
	Subsection 0	Tests Page Map status RAM parity logic.
	Subsection 1	Tests Page Map status RAM parity signal.
Section 45	Tests LMMIC function code parity logic.	
	Subsection 0	Tests LMMIC parity checker.
	Subsection 1	Tests CNTLP parity error signals.

Section 46	Tests the CAR Equal (FESE) comparator.	
	Subsection 0	Tests FESE comparator, Write/Read Delay, Compare Valid.
	Subsection 1	Tests FESE comparator, Write/Read Delay, Compare Invalid.
	Subsection 2	Tests FESE comparator, LRU Shortstop.
Section 47	Performs Conflict testing, allocation delays.	
	Subsection 0	Tests Conflict testing, Wait Allocate.
Section 48	Tests Cache/Page Map conflicts.	
	Subsection 0	Tests Cache to Map conflict (CHTMP).
	Subsection 1	Tests Read access and read/write conflict.
	Subsection 2	Tests Cache allocation after Word Not Valid Hit.
	Subsection 3	Tests Page Table update conflicts.
Section 49	Tests Invalidate conflicts.	
	Subsection 0	Tests Invalidate conflict, Write/Cache Hit.
	Subsection 1	Tests Invalidate conflict, Page Table Update.
	Subsection 2	Tests Invalidate conflict, Cache Lookahead.
	Subsection 3	Tests Invalidate conflict, Cache to Map.
	Subsection 4	Tests Invalidate conflict, Wait Allocate.
	Subsection 5	Tests Invalidate conflict, Cache Block Purge.
Section 50	Tests Function Code bit +Func(1).	
	Subsection 0	Tests Tests +Func(1) with CM-Exchange.
Section 51	Performs additional test for +MHIT signal.	
	Subsection 0	Tests Tests random read/write/allocate.
	Subsection 1	Performs final check of Page Map from Sub 0.

MAT3/P - Memory Array Identification Test

Section 0	Tests data retention ability with suppression of Refresh-related faults.	
	Subsection 0	Tests data retention ability of each array (or pair of arrays) in turn, with suppression of Refresh-related faults.
Section 1	Tests data retention ability without suppression of Refresh-related faults.	
	Subsection 0	Tests data retention ability of each array (or pair of arrays) in turn, without suppression of Refresh-related faults.
Section 2	Tests address lines by testing for address uniqueness.	
	Subsection 0	Tests address lines on each array (or pair of arrays) in turn, by testing for address uniqueness.
Section 3	Tests memory array with multiple accesses to each address, with random data.	
	Subsection 0	Tests each array (or pair of arrays) in turn, with multiple accesses to each testable address, with random data.
Section 4	Tests memory array with random data.	
	Subsection 0	Tests each array (or pair of arrays) in turn, with random data.
Section 5	Tests all of testable memory with random data and random addresses.	
	Subsection 0	Tests all of testable memory with random data and random addresses.

MCT3/P - MAC Test

Section 0	Performs initial MCH/MAC communication check.	
	Subsection 0	Tests that a Master Clear function, Activate Channel and Deactivate Channel receive proper responses from the MAC logic.
	Subsection 1	Tests that an Echo sequence operates correctly. Only one data pattern is used (FF_{16}).
	Subsection 2	Tests the basic MCH data path from the IOU to MAC and from MAC back to the IOU using the Echo function with various data patterns.
	Subsection 3	Tests the MAC control logic specifically related to clearing the Control Word 1 FF and Control Word 2 FF.
	Subsection 4	Tests the MAC control logic specifically related to clearing the Control Bytes Loaded FF.
	Subsection 5	Tests that a Clear Errors function sent to MAC receives proper responses from MAC logic.
Section 1	Performs initial check of MAC writing and reading capability.	
	Subsection 0	Tests that MAC will write directly accessible process registers (Type 0) without hanging up.
	Subsection 1	Tests that MAC will read all directly accessible process registers (Type 0) without hanging up.
	Subsection 2	Tests that MAC will write all Soft Control memories (Types 3, 4, 5 and 6) without hanging up.
	Subsection 3	Tests that MAC will read (Sweep) all Soft Control memories (Types 3, 4, 5 and 6) without hanging up.
	Subsection 4	Tests that MAC will write all locations in the register file (Type 7) without hanging up.
	Subsection 5	Tests that MAC will read all locations in the register file (Type 7) without hanging up.
	Subsection 6	Tests that MAC will write all of Control Store memory (Type 1) without hanging up.
	Subsection 7	Tests that MAC will read all of Control Store memory (Type 1) without hanging up.
	Subsection 8	Tests the clearing of the write FF by the function signal.
	Subsection 9	Tests the clearing of the read FF by the function signal.
	Subsections 10 through 15	Clock tests.

Section 2	Tests MAC data bus paths.	
	Subsection 0	Tests the MAC data bus path from the PFS Register to MCH for passing zero data, and tests that certain PFS Register bits are clear.
	Subsection 1	Tests the MAC data bus path within the MAC.
	Subsection 2	Tests the MAC data bus path from Address Control Soft Control 2 through the ALN Soft Control to the IF Decode RAM.
	Subsection 3	Tests the MAC bus data path legs from Register File to the BDP Port A and B, and from Address Control Soft Control 1 to the BDP RAM 0 and RAM 1.
Section 3	Tests PFS Register.	
	Subsection 0	Tests the byte select signals sent to and used on the PFS Register, as well as related control logic on the MAC.
	Subsection 1	Tests the setting of all bits in the PFS Registers by writing the PFS Registers with byte data with bit 7 set to a 1; causes all PFS Register bits to set.
	Subsection 2	Tests the clearing of PFS Register bits by writing the PFS Registers with byte data with bit 7 clear. All bits cannot be guaranteed to clear at this point in the testing sequence; a partial test.
	Subsection 3	Tests the PFS Select Translation logic and the Error Log Select signals sent to the PFS Register.
	Subsection 4	Tests that a Clear Errors Function clears the PFS Register bits; a partial test.
Section 4	Tests MAC hardware error reporting.	
	Subsection 0	Tests the MAC for reporting data errors on data received from the MCH.
	Subsection 1	Tests the MAC for reporting channel errors when illegal opcodes and illegal type codes are sent to MAC.
	Subsection 2	Tests the MAC parity checkers for reporting parity errors and checks that the MAC force bad parity circuitry functions properly (PTM bit 18).
	Subsection 3	Tests the parity error reporting on MAC bus data from PFS Register.
	Subsection 4	Tests the parity error reporting from PFS Register.
	Subsection 5	Tests the parity error reporting from PFS Register.
	Subsection 6	Tests the parity error reporting from PFS Register.
	Subsection 7	Tests the parity error reporting from PFS Register.

	Subsection 8	Tests the parity error reporting from the MAC Reference ROM read data parity checker.
	Subsection 9	Tests the MAC reporting of channel parity errors on Control Word 1, Control Word 2, and write data. Also ensures that these type parity errors are not reported as read data parity errors.
Section 5	Tests MAC hardware parity checkers.	
	Subsection 0	Tests the MAC parity checkers.
	Subsection 1	Tests the parity checkers on the PFS Register and the MAC Mux 5.
Section 6	Tests MAC reference ROM memory.	

NOTE

Clock pulse width margins (PARAM1) are enabled by section 6 for the remaining test sections.

	Subsection 0	Tests the MAC reference ROM memory for holding data in all locations.
	Subsection 1	Tests the MAC reference ROM read data parity checker.
	Subsection 2	Tests the reference ROM memory addressing logic, both external and internal to the memory's RAM chips.
Section 7	Tests MAC address bus path.	
	Subsection 0	Tests MAC address bus path with various address patterns.
Section 8	Tests AC/ALN SCM error reporting and byte selects.	
	Subsection 0	Tests parity error reporting from Address Control Soft Control 2 and MAC Bus Byte Select path to Address Control Soft Control 2.
	Subsection 1	Tests parity error reporting from Address Control Soft Control 1 and MAC Bus Byte Select path through Address Control Soft Control 2 to Address Control Soft Control 1.
	Subsection 2	Tests parity error reporting from ALN Soft Control and MAC Bus Byte Select path through Address Control Soft Control 1 to ALN SCM.

Section 9	Tests BDP RAM error reporting and byte selects.	
	Subsection 0	Tests RAM parity error reporting from the BDP Port A and MAC Bus Byte Select path through Address Control Soft Control 2 to the BDP Port A.
	Subsection 1	Tests RAM parity error reporting from the BDP Port B and MAC Bus Byte Select path through the BDP Port A to the BDP Port B.
	Subsection 2	Tests RAM parity error reporting from the BDP RAM 0, EBCDIC RAM, and the MAC Bus Byte Select path through the BDP Port B to the BDP RAM 0.
	Subsection 3	Tests RAM parity error reporting from the BDP RAM 0, Convert RAM.
	Subsection 4	Tests RAM parity error reporting from the BDP RAM 1, Spec Error RAM, and the MAC Bus Byte Select path through the BDP RAM 0 and 1.
	Subsection 5	Tests RAM parity error reporting from the BDP RAM 1, X256 RAM.
Section 10	Tests IF Decode RAMs error reporting and byte selects.	
	Subsection 0	Tests RAM parity error reporting from the IF C170 Decode RAMs.
	Subsection 1	Tests RAM parity error reporting from the IF C180 Decode RAMs.
Section 11	Tests MAC address bus error reporting.	
	Subsection 0	Tests parity error reporting from the IF Instruction Assembly Register (bits 0 through 7 and 8 through 15) parity checkers, from the BDP Port A and Port B Stage 1 Register parity checkers, and from the BDP RAM 0 and 1 Address Register parity checkers.
	Subsection 1	Tests the parity error reporting from the BDP Port A and Port B Stage 2 Register (bits 0 through 7) parity checkers.
	Subsection 2	Tests the parity error reporting from the BDP RAM 0 and 1 RAM Data Register (bits 0 through 7) parity checkers, EBCDIC and Spec Error RAMs.
	Subsection 3	Tests the parity error reporting from the BDP RAM 0 and 1 RAM Data Registers (bits 0 through 7) parity checkers, Convert and X256 RAMs.

Section 12	Tests IF C180 and C170 Decode RAM memories.	
	Subsection 0	Tests MAC address incrementer and parity predictor, and IF Instruction Assembly Register parity checkers for bits 0 through 7/64 and 8 through 15/65.
	Subsection 1	Tests IF C180 Decode RAM memory chips with various data patterns, all locations.
	Subsection 2	Tests IF C170 Decode RAM memory chips with various data patterns, all locations.
	Subsection 3	Tests the IF C180 Decode RAM read data parity checkers.
	Subsection 4	Tests the IF C170 Decode RAM read data parity checkers.
	Subsection 5	Tests the IF C180 Decode RAM addressing logic.
	Subsection 6	Tests the IF C170 Decode RAM addressing logic.
	Subsection 7	Tests the addressing circuitry internal to the IF C180 Decode RAM memory chips.
	Subsection 8	Tests the addressing circuitry internal to the IF C170 Decode RAM memory chips.
Section 13	Tests Address Control Soft Control 1 RAM memories.	
	Subsection 0	Tests Address Control Soft Control 1 memory chips with various data patterns, all locations.
	Subsection 1	Tests Address Control Soft Control 1 RAM read data parity checkers.
	Subsection 2	Tests Address Control Soft Control 1 addressing logic.
	Subsection 3	Tests the addressing circuitry internal to the Address Control Soft Control 1 memory chips.
Section 14	Tests Address Control Soft Control 2 RAM memories.	
	Subsection 0	Tests Address Control Soft Control 2 memory chips with various data patterns, all locations.
	Subsection 1	Tests Address Control Soft Control 2 RAM read data parity checkers.
	Subsection 2	Tests Address Control Soft Control 2 addressing logic.
	Subsection 3	Tests the addressing circuitry internal to the Address Control Soft Control 2 memory chips.

Section 15	Tests ALN SC RAM memories.	
	Subsection 0	Tests ALN SC memory chips with various data patterns, all locations.
	Subsection 1	Tests ALN SC RAM read data parity checkers.
	Subsection 2	Tests ALN SC addressing logic.
	Subsection 3	Tests the addressing circuitry internal to the ALN SC memory chips.
Section 16	Tests BDP Port A memory.	
	Subsection 0	Tests BDP Port A memory chips with various data patterns, all locations.
	Subsection 1	Tests BDP Port A memory read data parity checkers.
	Subsection 2	Tests BDP Port A addressing logic.
	Subsection 3	Tests the addressing circuitry internal to the BDP Port A memory chips.
Section 17	Tests BDP Port B memory.	
	Subsection 0	Tests BDP Port B memory chips with various data patterns, all locations.
	Subsection 1	Tests BDP Port B memory read data parity checkers.
	Subsection 2	Tests BDP Port B addressing logic.
	Subsection 3	Tests the addressing circuitry internal to the BDP Port B memory chips.
Section 18	Tests BDP RAM 0 EBCDIC memory.	
	Subsection 0	Tests BDP RAM 0 EBCDIC memory chips with various data patterns, all locations.
	Subsection 1	Tests BDP RAM 0 EBCDIC memory read data parity checkers.
	Subsection 2	Tests BDP RAM 0 EBCDIC memory addressing logic.
	Subsection 3	Tests the addressing circuitry internal to the BDP RAM 0 EBCDIC memory chips.
Section 19	Tests BDP RAM 0 Convert memory.	
	Subsection 0	Tests BDP RAM 0 Convert memory chips with various data patterns, all locations.
	Subsection 1	Tests BDP RAM 0 Convert memory read data parity checkers.
	Subsection 2	Tests BDP RAM 0 Convert memory addressing logic.

	Subsection 3	Tests the addressing circuitry internal to the BDP RAM 0 Convert memory chips.
Section 20	Tests BDP RAM 1 Spec Error memory.	
	Subsection 0	Tests BDP RAM 1 Spec Error memory chips with various data patterns, all locations.
	Subsection 1	Tests BDP RAM 1 Spec Error memory read data parity checkers.
	Subsection 2	Tests BDP RAM 1 Spec Error memory addressing logic.
	Subsection 3	Tests the addressing circuitry internal to the BDP RAM 1 Spec Error memory chips.
Section 21	Tests BDP RAM 1 X256 memory.	
	Subsection 0	Tests BDP RAM 1 X256 memory chips with various data patterns, all locations.
	Subsection 1	Tests BDP RAM 1 X256 memory read data parity checkers.
	Subsection 2	Tests BDP RAM 1 X256 memory addressing logic.
	Subsection 3	Tests the addressing circuitry internal to the BDP RAM 1 X256 memory chips.
Section 22	Tests Register File.	
	Subsection 0	Tests parity error reporting from the Register File read data parity checkers and the MAC Register File byte selection logic.
	Subsection 1	Tests the Register File with various data patterns, all locations, and the Control Store Disassembly Network byte selection logic.
	Subsection 2	Tests the Register File read data parity checkers and addressing logic external to the Register File memory chips.
	Subsection 3	Tests the addressing circuitry both external and internal to the Register File memory chips.
Section 23	Tests Control Store Breakpoint and Micrand Address Registers.	
	Subsection 0	Tests MAC capability to write and read the Control Store Breakpoint Register with various data patterns.
	Subsection 1	Tests the parity generator on the MAR/BKPT data path to MAC Mux2.
	Subsection 2	Tests MAC capability to write and read the Control Store Micrand Address Register with various data patterns, and test the Control Store Micrand address parity regeneration logic.

	Subsection 3	Tests parity error reporting from the Control Store Micrand Address Register parity checker.
	Subsection 4	Tests the Control Store Micrand Address Register parity checker.
Section 24	Tests Control Store memory.	
	Subsection 0	Tests parity error reporting from the Control Store memory data parity checkers and the MAC Control Store byte selection logic.
	Subsection 1	Tests the Control Store memory with various data patterns, all locations; the Control Store Micrand Address Incrementer logic; and the MAC Control Store read data path.
	Subsection 2	Tests the Control Store memory data parity checkers and partial test of the addressing logic.
	Subsection 3	Tests the Control Store memory addressing logic both external and internal to the Control Store memory chips.
Section 25	Completes test of MAC Byte Select logic.	
	Subsection 0	Completes testing of the MAC Byte Select signal formation for Type 4 Soft Control Memories (Address Control and ALN).
	Subsection 1	Completes testing of the MAC Byte Count translation logic.
	Subsection 2	Completes testing of the MAC Register File Byte Select Decoder logic.
Section 26	Completes test of MAC Type Decode logic.	
	Subsection 0	Completes testing of the MAC Type Decode logic.
Section 27	Completes test of MAC Go Decoder.	
	Subsection 0	Completes testing of the MAC Soft Control Go Decoder circuits involved with generation of the Go Soft Control memory signals.
	Subsection 1	Completes testing of the MAC BDP Control Memory Go Decoder circuits involved with generation of the Go BDP memory signals.
Section 28	Performs miscellaneous MAC logic testing.	
	Subsection 0	Tests MAC Data Mux4 and complete the testing of MAC Data Mux5.
	Subsection 1	Performs a partial test of MAC Data Mux1 and Data Mux2.

	Subsection 2	Tests that a Master Clear function to processor MAC generates Soft Control Go signals to the Address Control and ALN Soft Control memories, and function signal clears MAC Address Incrementer.
Section 29	Tests first failure capture feature.	
	Subsection 0	Tests first failure capture feature of system by forcing clock errors.
	Subsection 1	Restores system clock delays.

MDT3/P - ALN Multiply/Divide Test

Section 0 Checks ALN Soft Control memory addressing and Minor Cycle Control Register.

Subsection 0	Tests the Minor Cycle Control Register for parity status in Idle state, parity error detection and reporting, and Clock Enable controls from Soft Control memory.
Subsection 1	Tests Soft Control memory addressing from ALN micrand field and Minor Cycle clock.
Subsection 2	Tests Minor Cycle Control Register parity checkers and all bit paths into the register.

Section 1 Checks of C Register entry controls and sign extension.

Subsection 0	Tests C Register select of shifted data and sign extension from bit 0.
Subsection 1	Tests 32-bit sign extension in Divide Quotient Register and C Register entry from Divide Quotient Register.
Subsection 2	Tests C Sign Selection logic under 170/180 mode micrand control and its entry into the C Register.
Subsection 3	Tests Unsigned C Register entry from Major and Minor Cycle controls.

Section 2 Basic checks of multiply logic.

Subsection 0	Tests that the multiply logic does not pick up bits when multiplying zero times zero and one times zero.
Subsection 1	Tests the following areas of the multiply logic: B Register, B Product network for 1x decodes and partial sum generation, Multiply Partial Summing network for partial sum generation, the Final Adder and Multiply/Divide Output Mux.
Subsection 2	Tests the following areas of the logic: A Product network for 1x decode and partial sum generation, and the Multiply Partial Summing network for partial sum generation.
Subsection 3	Tests B Product network for partial carry generation and the Multiply Partial Summing network for partial sum generation.
Subsection 4	Tests the A Product network for partial carry generation and the Multiply Partial Summing network for partial sum generation.
Subsection 5	Tests B Product network for partial carry generation from 1x decodes in C Register bits 69 through 71 and 67 through 69.

Subsection 6	Tests A Product network for partial carry generation from 1x decodes in C Register bits 63 through 65 and 61 through 63.
Subsection 7	Tests B Product network for partial carry generation from 1x decodes in C Register bits 71 through 73 and 69 through 71.
Subsection 8	Tests A Product network for partial carry generation from 1x decodes in C Register bits 65 through 67 and 63 through 65.
Subsection 9	Tests B Product network for partial carry and partial sum generation.
Subsection 10	Tests A Product network for partial carry and partial sum generation.
Subsection 11	Tests Multiply Partial Summing network for summing of an A Product network partial sum with a B Product network partial sum.
Subsection 12	Tests Multiply Partial Summing network for summing of an A Product network partial sum with a B Product network partial carry.
Subsection 13	Tests the Multiply Partial Summing network for the summing of a Partial Carry from the A Product with a Partial Sum from the B Product network.
Subsection 14	Tests the Multiply Partial Summing network for Partial Carry 2 generation and the Multiply Final Adder for adding of Multiply Partial Carry bits.
Subsection 15	Tests the Multiply Partial Summing network for generation of N+2 carry.
Subsection 16	Tests the Multiply Partial Summing network for summing of a Multiply Partial Sum bit with a Net B Partial Carry bit.
Subsection 17	Tests the Multiply Partial Summing network for summing of a Multiply Partial Carry bit with a Net B Partial Sum bit.
Subsection 18	Tests 1x, −2x and Complement Correction decodes from C Register bits 65 through 73 to the B and A Product networks.
Subsection 19	Tests 1x, −2x and Complement Correction decodes from C Register bits 65 through 73 to the B and A Product networks and the Upper Multiply Partial Summing network.
Subsection 20	Tests 1x, −2x and Complement Correction decodes from C Register bits 61 through 67 to the A Product network.

	Subsection 21	Tests the 1x, −2x and Complement Correction decodes from C Register bits 61 through 67 to the A Product network and the upper bits of the Multiply Partial Summing Network.
	Subsection 22	Tests 2x decodes to the B Product network.
	Subsection 23	Tests the 2x decodes to the B Product network and the upper bits of the Multiply Partial Summing network.
	Subsection 24	Tests 2x decodes to the A Product network.
	Subsection 25	Tests the 2x decodes to the A Product network and the upper bits of the Multiply Partial Summing network.
	Subsection 26	Tests −1x decodes to the B Product network.
	Subsection 27	Tests the −1x decodes to the B Product network and the upper bits of the Multiply Partial Summing network.
	Subsection 28	Tests −1x decodes to the A Product network.
	Subsection 29	Tests the −1x decodes to the A Product network and the upper bits of the Multiply Partial Summing network.
	Subsection 30	Tests −1x decodes to the B Product network.
	Subsection 31	Tests the −1x decodes to the B Product network and the upper bits of the Multiply Partial Summing network.
	Subsection 32	Tests −1x decodes to the A Product network.
	Subsection 33	Tests the −1x decodes to the A Product network and the upper bits of the Multiply Partial Summing network.
	Subsection 34	Tests −0x decodes to the B Product network.
	Subsection 35	Tests the −0x decodes to the B Product network and the upper bits of the Multiply Partial Summing network.
	Subsection 36	Tests the −0x decodes to the A Product network.
	Subsection 37	Tests the −0x decodes to the A Product network and the upper bits of the Multiply Partial Summing network.
	Subsection 38	Tests C Register Sign control of complement decodes with decodes in C Register bits 67 through 73.
	Subsection 39	Tests C Register Sign control of complement decodes with decodes in C Register bits 61 through 67.
Section 3	Checks Multiply Final Adder logic.	
	Subsection 0	Tests Multiply Final Adder for Enable detection in the bit 7 position.
	Subsection 1	Tests Multiply Final Adder for Enable detection in the bit 6 position.

	Subsection 2	Tests Multiply Final Adder for Enable detection in the bits 5 through 0 positions.
	Subsection 3	Tests Multiply Final Adder Generate/Satisfy logic.
	Subsection 4	Tests Multiply Final Adder Generate/Enable/Satisfy logic.
	Subsection 5	Tests Multiply Final Adder Carry Interchange logic.
	Subsection 6	Tests Multiply Final Adder Logic for carry out of 12-bit iteration.
Section 4	Checks B Register entry controls.	
	Subsection 0	Tests B Register 32-bit sign selection and extension.
	Subsection 1	Tests B Register 48-bit entry, sign extended with bit 0 as sign.
	Subsection 2	Tests Signed/Unsigned Multiply B Register Entry controls. The A and B Product networks for disable of B Register bit 0 fanout and disable of B and C Sign to the Product network recoders.
	Subsection 3	Tests B Register entry of MSUM bits 8 through 71.
Section 5	Checks C Register entry and 170 multiply controls.	
	Subsection 0	Tests C Register for copies of C Register bits from ADATC.
	Subsection 1	Tests C Register entry of sign under 170 Mode Control.
	Subsection 2	Tests B Register Sign Selection and Entry under 170 Mode Control.
	Subsection 3	Tests Complement Correction signals for proper operation under 170 Mode Control.
Section 6	Checks C Register entry, multiply/divide control and sign signals to the ALN general network and the Major Cycle Control Buffer.	
	Subsection 0	Tests the entry and shifting of MSUM bits 60 through 71 through the C Register.
	Subsection 1	Tests Integer Multiply Overflow and the C Register and Multiply Final Adder zero checks.
	Subsection 2	Tests 32-bit Integer Multiply Overflow detection.
	Subsection 3	Tests multiply/divide unit for generation of Integer Multiply Divide Overflow and the ALN general network for encoding the overflow in the Normalize Encoder.
	Subsection 4	Tests B and C Operand Sign Selection logic to form Multiply/Divide Sign and the enable of the Multiply/Divide Sign to the B and C Sign Fanouts that are sent to the ALN general network.

	Subsection 5	Tests sequencing of the Multiply/Divide Sign to the Internal Sign in the Micrand Register and the enable of the Internal Sign to the C Sign Fanouts.
	Subsection 6	Tests C Register Bit 7 XORed with Multiply/Divide Sign and extended through the ALN General Network Shifter.
	Subsection 7	Tests Major Cycle Control Buffer for holding of U4, U6, and U8.
	Subsection 8	Tests Major Cycle Control Buffer for holding of U1, U2, and U3.
Section 7	Verifies operation of the Divide Quotient Register.	
	Subsection 0	Tests Divide Quotient Register for loading and its output through the C Register.
	Subsection 1	Tests Divide Quotient Register for ones complement and a 1 place shift.
	Subsection 2	Tests Divide Quotient Register for twos complement.
	Subsection 3	Tests the Major Cycle Hold Register for Integer Divide Control, U7.
	Subsection 4	Tests output of Divide Quotient Register for gating of Divide Fault and Divide Overflow.
	Subsection 5	Tests Divide Overflow for select of right shift one in the ALN general network.
Section 8	Checks Divisor and Remainder Register loading.	
	Subsection 0	Tests Divisor Register for 64-bit entry.
	Subsection 1	Tests Divisor and Remainder Registers for 48-bit entry.
	Subsection 2	Tests Divisor Register for 32-bit Integer entry.
	Subsection 3	Tests Remainder Register load from the Quotient Register.
	Subsection 4	Tests Divisor Register complement.
	Subsection 5	Tests Remainder Register for entry of Remainder Output.
Section 9	Checks Divide logic.	
	Subsection 0	Tests Divide logic and Carry Interchange.
	Subsection 1	Tests Divide Carry Interchange logic.
	Subsection 2	Tests Divide Round Control.
Section 10	Checks Integer Divide operation and Pop Counter.	
	Subsection 0	Tests Integer entry to the Divide Remainder Register.

	Subsection 1	Tests upper Divide.
	Subsection 2	Tests POP Counter.
Section 11		Checks multiply/divide unit for error status reporting, parity checking and parity generation.
	Subsection 0	Tests C Register parity error detection and reporting.
	Subsection 1	Tests Parity toggle on C Register shift data entry.
	Subsection 2	Tests C Register parity generators and checkers.
	Subsection 3	Tests B Register parity error detection and reporting.
	Subsection 4	Tests B Register parity checkers and Multiply Final Adder parity generators.
	Subsection 5	Tests Multiply Carry error status reporting.

OIT3/P - Operand Issue Test

Section 0	Tests functional unit micrand immediate data.	
	Subsection 0	Tests literal register address R.F. write logic.
	Subsection 1	Tests data path write width of 64 bits.
	Subsection 2	Tests data path write width of 48 bits.
	Subsection 3	Tests data path write width of 32 bits.
	Subsection 4	Tests data path write width of 16 bits.
Section 1	Tests register File read and Register File write.	
	Subsection 0	Tests RDSb Register File read address logic.
	Subsection 1	Tests RDSb Data path to ALN.
	Subsection 2	Tests RDSc Register File read address logic.
	Subsection 3	Tests RDSc Data path to ALN.
Section 2	Tests Operand Left Shift network (result to ALN).	
	Subsection 0	Selects rightmost 32 Bits *2.
	Subsection 1	Selects rightmost 32 Bits *8.
	Subsection 2	Selects rightmost 18 Bits *8.
Section 3	Tests 60-bit sign extension.	
	Subsection 0	Tests sign extension on B Operand (SEb Field).
	Subsection 1	Tests sign extension on C Operand (SEc Field).
Section 4	Tests Register File read of literal RDS data.	
	Subsection 0	Tests RDSb literal data path to ALN.
	Subsection 1	Tests RDSc literal data path to ALN.
	Subsection 2	Tests RDSa literal data path to Address Control.
Section 5	Tests Live Register write/read.	
	Subsection 0	Tests write/read UCR.
	Subsection 1	Tests write/read MCR.
Section 6	Tests Live Register Write/Read.	
	Subsection 0	Tests Write/Read SIT.
	Subsection 1	Tests Write/Read PIT.
Section 7	Tests operand shift logic (result to Address Control).	
	Subsection 0	Select rightmost 18 bits *8; DS = 3.

Section 8	Tests Largest Ring Check and Selection logic.	
	Subsection 0	Tests data Mux to PVA Register (SM) Path through Largest Ring Select.
	Subsection 1	Tests DAI to Register File Path through Ring Select (WW = 1).
	Subsection 2	Tests DAI to Register File Path through Ring Select (WW = 3).
Section 9	Tests Load Store Multiple Counters.	
	Subsection 0	Tests N Counter.
	Subsection 1	Tests As Counter.
	Subsection 2	Tests Xs Counter.
	Subsection 3	Tests As/Xs parity logic.
Section 10	Tests Load Store Multiple Counters.	
	Subsection 0	Tests At using CC = 2 Load.
	Subsection 1	Tests Xt using CC = 2 Load.
	Subsection 2	Tests At using CC = 3 Load (also at< 2 Instruction Specification error).
	Subsection 3	Tests At using CC = 7 Load (also at < 2 Environment Specification error).
	Subsection 4	Tests At/Xt parity logic.
Section 11	Tests Immediate Operand Mux 1.	
	Subsection 0	Tests Immediate Operand D input.
	Subsection 1	Tests Immediate Operand K input.
	Subsection 2	Tests Immediate Operand jk (C170) input.
	Subsection 3	Tests Immediate Operand jk (C180) input.
	Subsection 4	Tests Immediate Operand j (C180) input.
	Subsection 5	Tests Immediate Operand jkQ input.
	Subsection 6	Tests Immediate Operand Q input.
	Subsection 7	Tests C170 RAC Input.
Section 12	Tests address Control Immediate Data Mux (IDS Mux).	
	Subsection 0	Tests Zero Data Constant generation.
	Subsection 1	Tests constant = 8 hexadecimal generation.
	Subsection 2	Tests D Zero Extended input.

	Subsection 3	Tests Q Sign Extended input.
	Subsection 4	Tests 8*D Zero Extended input.
	Subsection 5	Tests 8*Q Sign Extended input.
	Subsection 6	Tests 8*K Sign Extended input.
	Subsection 7	Tests RDSB Save Register.
Section 13	Tests RAF write with RDS Fixed (RDSa).	
	Subsection 0	Tests RAF = 0 (j).
	Subsection 1	Tests RAF = 1 (k).
	Subsection 2	Tests RAF = 2 (170i).
	Subsection 3	Tests RAF = 4 (j+1).
	Subsection 4	Tests RAF = 5 (k+1).
	Subsection 5	Tests RAF = 6 (180i).
Section 14	Tests RAF read with RDS Fixed (RDSc).	
	Subsection 0	Tests RAF = 0 (j).
	Subsection 1	Tests RAF = 1 (k).
	Subsection 2	Tests RAF = 2 (170i).
	Subsection 3	Tests RAF = 4 (j+1).
	Subsection 4	Tests RAF = 5 (k+1).
	Subsection 5	Tests RAF = 6 (180i).
Section 15	Tests RAF with RDS Varied.	
	Subsection 0	Tests RAF during Register File Write using RDSd.
	Subsection 1	Tests RAF during Register File Read using RDSa.
	Subsection 2	Tests RAF during Register File Read using RDSb.
Section 16	Tests RDS Bit 4 (OR bit).	
	Subsection 0	Tests OR Bit during Register File Write using RDSa.
	Subsection 1	Tests OR Bit during Register File Write using RDSd.
	Subsection 2	Tests OR Bit during Register File Read using RDSa.
	Subsection 3	Tests OR Bit during Register File Read using RDSb.
	Subsection 4	Tests OR Bit during Register File Read using RDSc.
Section 17	Tests RDS BI Bits (bits 2 and 3).	
	Subsection 0	Tests BI Bits in RDS during write using RDSa.

	Subsection 1	Tests BI Bits in RDS during write using RDSd.
	Subsection 2	Tests BI Bits in RDS during read using RDSa.
	Subsection 3	Tests BI Bits in RDS during read using RDSb.
	Subsection 4	Tests BI Bits in RDS during read using RDSc.
Section 18	Tests UZ check logic (no conflict).	
	Subsection 0	Tests UZ check logic using RDSc.
	Subsection 1	Tests UZ check logic using RDSb.
	Subsection 2	Tests UZ check logic using RDSa.
Section 19	Tests sign extend memory data (SEd) and Zero Fill RDSc logic (ZF).	
	Subsection 0	Tests logic controlled by SEd.
	Subsection 1	Tests Zero Fill RDSc logic (ZF).
Section 20	Tests parity logic; generators and associated checkers.	
	Subsection 0	Tests parity network at Register File Address Minipipe.
	Subsection 1	Tests parity network having generator at Literal Data input to Immediate Operand Mux 1.
	Subsection 2	Tests parity network having generator at RDSw input to ICP.
	Subsection 3	Tests parity network at Microsecond Counter.
Section 21	Tests parity logic; detection and reporting of errors.	
	Subsection 0	Tests parity network at Register File Address Minipipe.
	Subsection 1	Tests parity network having generator at Literal Data input to Immediate Operand Mux 1.
	Subsection 2	Tests parity network having generator at RDSw input to ICP.
	Subsection 3	Tests parity network at Microsecond Counter.
Section 22	Tests parity networks having checkers in OPI; detection/reporting errors.	
	Subsection 0	Tests RDSw parity logic.
	Subsection 1	Tests Virtual Instruction data parity logic.
Section 23	Tests block Register File write.	
	Subsection 0	Tests block Register File write on DAI parity error.
Section 24	Tests Minipipe comparators (no compare).	
	Subsection 0	Tests Rank 40 and Rank 50 comparators.

PDT3/P - Processor Detected Malfunction and Associated Logic

Section 0	Tests MCR and UCR.	
	Subsection 0	Tests the MCR and UCR with live register writes.
Section 1	Tests MCR.	
	Subsection 0	Tests Outward Call/Inward Return.
	Subsection 1	Tests Invalid Segment/Ring 0.
	Subsection 2	Tests SIT.
	Subsection 3	Tests Page Table Search without Find.
	Subsection 4	Tests access violation.
	Subsection 5	Tests address specification error.
	Subsection 6	Tests instruction specification error.
	Subsection 7	Tests environment specification error.
Section 2	Tests detected uncorrectable error.	
	Subsection 0	Tests Address Control before PONR PDM.
	Subsection 1	Tests general micrand parity fault.
	Subsection 2	Tests Load Register parity fault.
	Subsection 3	Tests Tests IF PDM.
	Subsection 4	Tests Tests BDP Convert Error to MCR(0).
	Subsection 5	Tests BDP Input PE to MCR(0).
Section 3	Tests UCR.	
	Subsection 0	Tests Invalid BDP data.
	Subsection 1	Tests Arithmetic Loss of Significance.
	Subsection 2	Tests FP Indefinite.
	Subsection 3	Tests FP Loss of Significance.
	Subsection 4	Tests Exponent Underflow.
	Subsection 5	Tests Exponent Overflow.
	Subsection 6	Tests Arithmetic Overflow.
	Subsection 7	Tests Divide Fault.
	Subsection 8	Tests Inter-ring Pop.
	Subsection 9	Tests PIT.

Section 4	Tests C170 Exit Mode Register.	
	Subsection 0	Tests C170 Indefinite Operand.
	Subsection 1	Tests C170 Infinite Operand.
	Subsection 2	Tests C170 Operand AOR.
	Subsection 3	Tests C170 RNI AOR.
	Subsection 4	Tests C170 Illegal Instruction.
	Subsection 5	Tests C170 address correct if held up.
Section 5	Tests parity detection logic.	
	Subsection 0	Tests P Register parity detection in ICP.
	Subsection 1	Tests UTP parity detection in ICP.
	Subsection 2	Tests PDM on Page Table errors.

SMT3/P - Segment Map Test

Section 0	Tests basic paths.	
	Subsection 0	Tests the PVA input via parity status.
	Subsection 1	Tests setting of the UVMID flag.
	Subsection 2	Tests the Segment Descriptor Mux, Gate 3.
	Subsection 3	Tests the PVA register, New P Register and Old P Register.
	Subsection 4	Tests the Old P Register input to the Global Key Input Mux.
Section 1	Tests the ring compare logic.	
	Subsection 0	Tests the R2 ring compare logic.
	Subsection 1	Tests the largest ring compare logic.
	Subsection 2	Tests the Old P RN input to the R1/PVA RN Mux and the PVA RN input to the PVA/OLD P RN Mux.
	Subsection 3	Tests the Ring Hold Register and the ring hold compare logic.
	Subsection 4	Tests the inter-ring compare logic.
Section 2	Tests segment validity logic.	
	Subsection 0	Tests segment validity logic via a CALL1 microfunction.
	Subsection 1	Tests segment validity.
	Subsection 2	Tests global key/lock test logic.
	Subsection 3	Tests the local key/lock test logic.
Section 3	Tests map read/write support logic.	
	Subsection 0	Tests the monitor/job status RAMs.
	Subsection 1	Tests the segment map set 0,1 hit compare logic.
	Subsection 2	Tests the set 0,1 write select translator logic.
	Subsection 3	Tests the status RAM update control logic.
	Subsection 4	Tests the status RAMs for address uniqueness.
Section 4	Tests the Map sets.	
	Subsection 0	Tests the segment number portion of Map set 0 and 1.
	Subsection 1	Tests the Map set 0,1 RAM bits 0 through 47 and parity.

Section 5	Tests segment table address.	
	Subsection 0	Tests the segment table address register.
	Subsection 1	Tests the segment table address save register.
	Subsection 2	Tests the status RAM purge control logic.
	Subsection 3	Tests the segment table length test logic.
Section 6	Tests PDMs generated by segment Map.	
	Subsection 0	Tests the SMPDM with NEW P, VMID and CBP R3 parity errors.
	Subsection 1	Tests the SMPDM by forcing STL parity errors.
	Subsection 2	Tests the SMPDM by forcing micrand parity errors.
	Subsection 3	Tests the SPDMBP by forcing Segment Descriptor Mux parity errors.
	Subsection 4	Tests the SPDMBP by forcing PVA parity errors.
	Subsection 5	Tests the SPDMBP by forcing Status RAM errors.
	Subsection 6	Tests the Disable PDM's control in Segment Map.
Section 7	Tests miscellaneous logic.	
	Subsection 0	Tests SM control of LM purges.

Operator Communication 3

- Displays . . . 3-1
  - Running Display . . . 3-1
  - Set Parameters Display . . . 3-2
  - Error Message Display . . . 3-3
  - CMT3P Error Displays . . . 3-4
  - Isolation Displays . . . 3-4
    - Normal Isolation Display . . . 3-4
    - Fault Not Isolatable Display . . . 3-5
    - Informational Display . . . 3-5
    - Isolation Diagnostic Internal Error Display . . . 3-5

- Operator Entries . . . 3-6
  - Running Procedures . . . 3-6
    - Load and Run the Test . . . 3-6
      - Change Test Parameters . . . 3-8
      - Stop on Error . . . 3-8
      - Report individual Condition Errors . . . 3-8
      - Loop on Test . . . 3-9
      - Loop on Section . . . 3-9
      - Loop on Subsection . . . 3-9
      - Loop on Iteration . . . 3-10
      - Loop on Condition . . . 3-10
      - Run a Scope Loop . . . 3-10
      - Run with Wide or Narrow Clock Pulse Width Margin . . . 3-11

- Normal Messages . . . 3-11

- Error Messages . . . 3-11

- Error Codes . . . 3-14
  - BASICTC/DEX Error Codes . . . 3-14
    - Program Internal Fault Error Codes . . . 3-15
    - External Fault Error Codes . . . 3-15
  - Test Error Codes . . . 3-16

- Applications . . . 3-17

Operator Communication 3

This section explains displays, operator entries, messages, and error codes.

Displays

This subsection explains and gives examples of the following displays.

- Running displays
- Parameter displays
- Error message displays
- Isolation (Fault) displays

Running Display

The standard running display has the following format:

```
name   op    PCxxxx Sxxxx SBxxxx Cxxxx Ixxxx VER=yy/mm/dd
```

Element	Description
name	Test name.
op	Type of activity.
RU	Running message.
SC	Stopped at end of condition.
SB	Stopped at end of subsection.
SI	Stopped at end of iteration.
SS	Stopped at end of section.
ST	Stopped at end of test.
SE	Stopped on error.
RC	Repeating condition.
RB	Repeating subsection.
RI	Repeating iteration.
RS	Repeating section.
RT	Repeating test.
SM	Scope mode.
HT	Test has been stopped by S key entry.
PCxxxx	Pass count in decimal.
Sxxxx	Current section number in decimal.
SBxxxx	Current subsection number in decimal.
Cxxxx	Current condition number in decimal.
Ixxxx	Current iteration in hexadecimal, within the subsection being run.
VER	Test version date (year, month, day).

Except when the operation is SE, the running display is bypassed when the bypass all messages bit (PARAMO, bit 2^{12}) is set, or the display only error messages bit (PARAMO, bit 2^{13}) is set, or the scope mode bit (PARAMO, bit 2^{10}) is set.

When the operation is RU, the SB, I, and C values are not dynamic. That is, their values are those at the time the test sent the RU message, which is normally at the beginning of a section when SB and C are 0000.

When the operation is ST, the S value is not definable.

Set Parameters Display

The set parameters display appears at the beginning of the test to allow the customer engineer to set test parameters. After the desired parameters are set, a space keyboard input will cause the test to continue. This display is bypassed when the bypass parameter stop bit is set (PARAM1, bit 2^2).

The following is an example of the set parameters display:

```
name SET PARAMS  PA=0122B  yy/mm/dd

 THIS IS A P3 DETECTION TEST

 OVERLAYS  = Y   MARGINS   = Y
 RAM H/W   = Y   ISOLATE   = N
 MICRCODE  = Y


 SECTIONS =
 00-09  16  20--24
```

where:

name = Test name.
PA = First word address of parameter area in PP memory in octal.
yy/mm/dd = Year/month/day (test version date).
P3 = Processor element identifier.
Y = Yes; indicated option is selected.
N = No; indicated option is not selected.
SECTIONS = Sections selected. Only one line is reserved for this portion of the display. Should the number of sections selected exceed the line capacity, the message SEE PARAM5 THRU PARAM8 is displayed instead.

NOTE

Sections indicated are the set conditions of the bits in PARAM 5, 6, 7, and 8 regardless of whether such sections actually exist.

Error Message Display

The error message display is shown when an error is detected. The error display is bypassed when the bypass all messages bit (PARAMO, bit 2^{12}) is set or when scope mode bit (PARAMO, bit 2^{10}) is set.

Test execution stops when stop on error (PARAMO bit 2^4) is set. A keyboard input of space causes test execution to resume.

```
 name   SE   PCxxxx   Sxxxx   SBxxxx   Cxxxx   Ixxxx VER=yy/mm/dd

EC1=xxxx   EC2=xxxx   TE=xxxx   FSC1=ABBCCDD   FSC2=EEEEFF

error message

nn EXP= zzzz            RCV= zzzzz          F
```

where:

name = Test name.
SE = Error stop.
PCxxxx = Pass count in decimal.
Sxxxx = Failing section number in decimal.
SBxxxx = Failing subsection number in decimal.
SCxxxx = Failing condition number in decimal when reporting a condition error. It is the condition number of the last condition in a failing subsection when reporting a subsection error.
VER = Test version date (year, month, day).
EC1 = Refer to Error Codes for definition.
EC2 = Refer to Error Codes for definition.
TE = Total error count in decimal.
FSC1 = Fault symptom code 1 for reported error.

    A = Test that failed.

        1 = ACT3P   2 = ANT3P   3 = BPT3P   4 = CMT3P   5 = CST3P
        6 = CTT3P   7 = ICT3P   8 = IFT3P   9 = LMT3P   A = MAT3P
        B = MCT3P   C = MDT3P   D = OIT3P   E = PDT3P   F = SMT3P

    BB = Section number.
    CC = Subsection number.
    DD = Iteration number (lower 8 bits).

FSC2 = Fault symptom code 2 (Extension of FSC1).

    EE = Failing conditions (C.15-C.0; C.0 right justifies).
    FF = Bytes miscompared in 1st bad condition (Byte 0 - Byte 7).

error message = Refer to Error Messages for definitions of messages that can appear here and on the following line.
Ixxxx = Failing number (xxxx) in hexadecimal of the iteration within the subsection which failed.
nn = Test condition number.
zzzz = Expected/Received test result for the specified condition.
F = F is indicated when error is detected; otherwise, no F appears.

CMT3P Error Displays

In CMT3P, all addresses are expressed in interleaved mode.

All maintenance register content reports are the same as the format presented by the maintenance register unless stated otherwise.

Isolation Displays

In addition to the preceding displays, when isolation is selected (Bit 2^6, PARAM1) the following isolation displays are provided.

Other test parameters have no control over whether or not these displays appear (always displayed). If the error display is being logged, the isolation display will also be logged.

Normal Isolation Display

```
name   SE   PC000X   S00ss   SB00bb   C000X    VERSION   yy/mm/dd
ISOLATION FROM TREE nnnn

(optional message)

FAILING MODULE OR CONNECTOR ISOLATED TO THE FOLLOWING
LOCATIONS IN ORER OF DECREASING PROBABILITY

1.  LOCATION 111111    TYPE xtty
2.  LOCATION 111111    TYPE xtty
3.  LOCATION 111111    TYPE xtty
       .        .         .  .   (from one to nine lines are
       .        .         .  .   displayed.)
       .        .         .  .
9.  LOCATION 111111    TYPE xtty
```

where:

name	Test name.
PC000X	Pass count. Always X for isolation.
S00ss	Number of section that failed.
SB00bb	Number of failing subsection within the failing section.
C000X	Condition number. Always X for isolation.
VERSION	Version date of this isolation diagnostic.
optional message	An isolation diagnostic information message may or may not be provided.
111111	Physical subassembly location in the cabinet containing the hardware being tested by the test.
TYPE xtty	Subassembly type designator. Only the two middle characters of the normal 4-character subassembly type designator are used. When different type subassemblies can be used in the same location, the displayed type code will normally contain an x for the characters that vary (that is, location B21 can contain EA or EB type boards. Displayed type will be xExy). x is a year; y is a division designator.
nnnn	Name of the code supplying the isolation logic. This is CMK3 for the 96X model 34 memory.

NOTE

Because of nonstandard methods used for ECO changing of subassembly type designators, it is possible for the isolation diagnostic to list a board type designator that does not agree with the board type found in the called-out location. The location will be correct.

In addition to the paks called out, it is possible that a clock pak has failed. Refer to Special Maintenance Aids in the Maintenance and Parts Manual for a listing of clock paks and the pak locations they feed.

Fault Not Isolatable Display

```
name    SE    PC000X    S00ss    SB00bb    C000X    VERSION    yy/mm/dd
ISOLATION FROM TREE nnnn
FAULT NOT ISOLATABLE.  (optional message)
```

All variables are the same as those defined for the normal isolation display.

FAULT NOT ISOLATABLE means that the isolation diagnostic cannot isolate the test failure to a specific set of suspected faulty subassemblies. The optional message will usually provide useful information about the failure.

Informational Display

```
name    SE    PC000X    S00ss    SB00bb    C000X    VERSION    yy/mm/dd
ISOLATION FROM TREE nnnn

(one-line informational message)
```

Isolation Diagnostic Internal Error Display

```
name    SE    PC000X    S00ss    SB00bb    C000X    VERSION    yy/mm/dd
ISOLATION FROM TREE nnnn

(TREE EXECUTION ERROR xx at ADDRESS yyyy
```

All variables in the top line of the display are the same as those defined for the normal isolation message.

The last displayed line indicates that an error has occurred during the execution of the isolation diagnostic decision tree function codes.

yyyy	PP address of the decision tree function code that caused the error.
xx	Error code identifying the type of error.

00	A call to CMSE by the isolation diagnostic was rejected by CMSE. The address is meaningless.
02	An isolation diagnostic overlay is missing on the library (CMSE could not find the requested overlay).
03	An error occurred when CMSE was loading an isolation diagnostic overlay.
10	The function code at address yyyy does not conform to any of the legal decision tree function codes.
20	An attempt was made to execute a decision tree function code from an address (yyyy) that is not in the allowable areas for such function codes.
30	The function code at yyyy has illegal parameter values.

Error codes 10, 20, and 30 are definitely caused by bugs in the isolation diagnostic and should be reported by PSR accompanied by copies of the corresponding test error display and isolation display.

Error codes 00, 02, and 03 can be caused by a bad library (all diagnostic overlays were not loaded on the library device); bad IOU, disk or other hardware; or a bug in the isolation diagnostic.

Operator Entries

Fault identification tests use standard CMSE commands as described in the MSL 15X Reference Manual.

Running Procedures

To use an identification test for hardware error detection, the following operating procedures should be used. These procedures assume that CMSE is already running.

Load and Run the Test

Command	Function
DP,num(CR)	Deadstarts the PP (num) in which test will run. Loads PP with CMSE resident code and starts the PP executing idle loop.
CF(CR)	Tells CMSE to stop monitoring CM.
CX,M	Master clear memory.
CX,P	Master clear processor.

CP,num,test name(CR)	Loads PP num with the test from the maintenance library device.
AN(CR)	Turns off A screen display. This results in shorter test run time.

NOTE

False errors may be reported if central memory and/or maintenance registers are displayed by CMSE while the test is executing and forcing hardware errors.

BP,num(CR)	Assigns the B display to the PP running the test (PP num). This allows the test to display test messages.
RU,num,100(CR)	Initial start of test execution in PP num at location 100. To restart the test, type R (CR).
S(CR)	Stops test execution. This is interpreted as a stop command. That is, DEX goes into a loop, maintaining communication, but no test activity occurs. Control is returned to the test upon receipt of a keyboard entry of a space or any character other than S or R.
(space)	Resumes test execution from point at which it stopped. The existing display at the time the space bar is pressed is blanked out to indicate acceptance of the command by the program.

NOTE

Entry of any character other than S or R is also interpreted as a space command.

R(CR)	Restart test from beginning. The current pass count is not altered.
DP,num(CR)	Stops test execution in PP specified by num and idles the PP. To restart the test, the user must reload the test code and type RU,num,100(CR).

NOTE

When all sections have been executed and the repeat test parameter is not selected, the test relinquishes control completely. In order to rerun the test, it must be reloaded and rerun.

LT,num,test binary name(CR)	Load and execute the test.

Change Test Parameters

The test will stop after the RU command with a set parameters message displayed. Refer to Set Parameters Display.

The customer engineer can then change the default parameter selection, if desired. Refer to Parameters and Control Words for definitions of parameter bits and default settings.

Command	Function
Cxx or Sxx(CR)	Clear/Set xx bit of PARAM0 or PARAM1, where xx is the bit mnemonic such as ST, SS, and so forth. (Refer to PARAM0 and PARAM1 description in Parameters and Control Words mnemonic definitions.)
EP,num,adrs,data(CR)	Enter parameter data into PP num at location adrs. EP commands must be used for changing parameter words other than PARAM0, but can also be used to change PARAM0.
(space)	Causes the test to resume execution. In this case, the test starts execution using the newly set parameters or the default parameters if no parameter changes were made.

NOTE

Ensure that PARAM4, bits 2^8 through 2^{11}, contain the proper maintenance channel radial connect code for the equipment being tested.

Stop on Error

Set stop on error (SSE) when entering test parameters or whenever the test is stopped. The test will then stop upon detecting a hardware failure and display pertinent error information, when bypass all messages (PARAM0 bit 2^{12}) is not selected. Refer to Error Messages.

Under default operation of the test, errors are reported at the end of the iteration in which one or more of the conditions failed within an iteration.

The failing iteration condition checks (those that failed and those that did not fail) provide multisymptom failure data about a single failing hardware element. This aids the fault isolation process.

Typing in a space will resume test execution.

Report individual Condition Errors

Set report condition error (SCE), PARAM1 bit 2^8, when entering test parameters or whenever the test is stopped. The test will report each condition that fails, when bypass all messages, PARAM0 bit 2^{12}, is not selected.

When stop on error is selected, the test will also stop when reporting condition errors.

Typing in a space will resume test execution.

Loop on Test

There are three ways to loop on the test:

1. Set repeat test (SRT) when entering test parameters or whenever the test stops. The test will then repeat when it reaches the end of test. Remove all stop parameter selections if continual test looping without stops is desired.
2. Enter PARAM2 with a repeat test count when entering test parameters or whenever the test stops. The test will execute the number of times specified by PARAM2 and then exit, if repeat test parameter bit is not set. All stop parameter bits must be clear if continual looping, without stopping until exit, is desired.
3. Enter PARAM18 with a repeat subsection count when entering test parameters. Each subsection will then be repeated the number of times specified by PARAM18 before executing the next sequential subsection. The test will exit to CMSE idler when the last subsection of the last selected section has been executed, if repeat test parameter bit is not set.

Typing in an S(CR) will stop test execution. Typing in a space will resume test execution.

Loop on Section

There are two ways to loop on a section:

1. When entering test parameters, select the desired section to be executed (only one section should be selected) and set repeat section (SRS). The selected section will then be continuously executed. Stop at end of section, subsection, and condition parameter bits must be clear for nonstop section looping.
2. Select repeat section by typing SRS any time test execution stops, other than at the end of test. The current section will be repeated continually when test execution is resumed. Stop at end of section, subsection, and condition parameter bits must be clear for nonstop section looping.

Typing in an S(CR) will stop test execution. Typing in a space will resume test execution.

Loop on Subsection

There is one way to loop on a subsection:

> Set the repeat subsection parameter bit by typing SRB(CR) whenever the test stops on an error stop, end of condition stop, or end of subsection stop. The current subsection will be repeated continually when test execution is resumed. Stop at end of subsection and condition parameter bits must be clear for nonstop subsection looping.

Typing in an S(CR) will stop test execution. Typing in a space will resume test execution.

Loop on Iteration

There is one way to loop on an iteration:

> Set the repeat iteration parameter bit by typing SRI(CR) whenever the test stops on an error stop, end of condition stop (SC), or end of iteration (SI) stop. The current iteration (test sequence which includes all conditions necessary to recreate the error) is repeated. This will allow repeating multi-iteration sequences within a subsection where repeat conditions will not recreate the error. Stop at end of iteration and condition parameter bits must be clear for nonstop iteration looping. Reference individual test write-ups to determine the location of the repeat iteration loop.

Typing in an S(CR) will stop execution. Typing in a space will resume test execution. Typing in SAB (set abort subsection) will abort to the end of the subsection.

Loop on Condition

There is only one way to loop on a condition check. Set the repeat condition parameter bit by typing SRC(CR) whenever the test stops on a condition error stop (stop on error parameter bit SE set and report condition errors PARAM1 bit 2^8 set) or an end of condition stop. The current condition will be continually repeated when test execution is resumed. Stop at end of condition parameter bit must be clear for nonstop condition looping.

NOTE

It is not always possible to recreate the error by looping on a condition. This is because a prior condition may have set up the error and is needed for the loop. For these types of errors use repeat iteration (SRI) which runs all the subsection conditions only for the failing pattern.

To improve timing, ten repeats are done before the display controls are sampled.

Typing in an S(CR) will stop test execution. Typing in a space will resume test execution.

Run a Scope Loop

The test can be put in scope mode. Scope mode results in the selected condition/iteration being repeated without any communication to CMSE and minimum execution of other overhead code external to the condition code. Refer to section 4.6, Applications, for multioperand/result subsections.

Scope mode should be selected only after a condition/iteration error stop or after an end-of-condition/end-of-iteration stop, using the following procedure:

1. Set repeat condition/iteration by typing SRC(CR)/(SRI(CR).
2. Set stop at end of condition/iteration by typing SSC(CR)/SRI(CR).
3. Set scope mode by typing SSM(CR).
4. Blank the left screen by typing AN(CR).
5. Press space bar.

NOTE

The test will continually repeat the present condition/iteration code. There will be no communication from the test during scope mode.

NOTE

If the test is at a condition/iteration error stop when scope mode is selected, the space key will have to be repeated at the end of the condition stop.

To exit from scope mode and regain control of the test, type:

```
UP,xx
```

Where xx is the number of the PP in which the test is executing. The test now communicates with the operator in the normal manner, by displaying a condition/iteration error message or an end-of-condition/end-of-iteration stop message. The scope mode parameter bit is also cleared.

Run with Wide or Narrow Clock Pulse Width Margin

Set wide clock margin (PARAM1 bit 2^0) or narrow clock margin (PARAM1 bit 2^1) when entering the test parameters. Also ensure that PARAM16 bit 2^0 is clear.

A few tests will change the clock margins regardless of the setting of PARAM16 bit 2^0. Refer to Restrictions and User Cautions and to individual test descriptions.

If both wide clock margin and narrow clock margin are selected at the same time, only narrow clock margins will be enabled by the hardware.

Margins can be set using CMSE commands prior to running the test(s). This is the recommended practice.

Normal Messages

Other than the running display and set parameters display described previously, fault identification tests do not display any test messages for normal operation.

Error Messages

All test messages are displayed on the B display. The B display must be assigned to the PP in which the test is executing before the test messages will be displayed.

NOTE

Dropping the test's display by assigning the B display to a different PP or to Central Memory will cause the presently displayed test message to be lost. Assigning the B display back to the test's PP will result in a blank display until the test displays a new message. New messages will not be displayed if the test was stopped before the display was dropped. In the latter case (B display reassigned to test's PP but test is stopped), typing in a space will resume test execution.

When a fault identification test detects an error and the error message display appears, one of the following messages might be shown in the third and possibly fourth lines of the display.

Message	Explanation
INTERNAL FAULT BAD BASICTC CALL	The test called one of the basic test control sub routines using improper call parameters. This is an unrecoverable fault. Refer to BASICTC/DEX Error Codes for definitions of the error codes.
EXTERNAL FAULT COULD BE MCH/CMSE/ETC.	The test requested a function on an external device (for example, the maintenance channel, CMSE, and so on) and the function was not performed or was performed incorrectly. This is an unrecoverable fault. Refer to BASICTC/DEX Error Codes for definitions of the error codes.
COND ERR or COND ERR--rrrr ERROR IN pppp qqq	This message is only displayed when a condition check fails and report condition errors PARAM1 bit 2^8 (SCE) is set. Refer to BASICTC/DEX Error Codes for definitions of the error codes. The failing condition number, iteration number, subsection number, and section number are in the error display header. The condition number (nn), expected results (EXP=x-x), received results (RCV=x-x) are displayed on the fourth line. An F indicates that the condition check failed. The longer error message is only displayed if bypass error code 1 crack (bit 2^9 in PARAM1) is clear. In this case, the value in CW4 (error code 1) is cracked to provide the following: rrrr = type of failure (for example, DATA) pppp = element (for example, CPU1) qqqq = functional area of element (for example, CACHE) Example: COND ERR--DATA ERROR IN CPU1

Message	Explanation
ITER ERR or ITER ERR--rrrr ERROR IN pppp qqqq	The hardware being tested by the subsection code did not perform as expected (one or more of the condition checks within one of the subsections iterations failed). Refer to BASICTC/DEX Error Codes for definitions of the error codes. The iteration conditions (nn) are displayed on the fourth and subsequent display lines. Each line will include the condition number (nn), expected results (EXP=x-x for first condition, x-x for subsequent conditions) and received results (RCV=x-x for first condition, x-x for subsequent conditions). F flags indicate those conditions which failed. Refer to COND ERR for a description of rrrr, pppp, and qqqq. The CE can select any of the following options: 1. Attempt to duplicate the error results by typing in x(CR), where x is the number of times (1 through 9) the failing subsection should be rerun. If duplicate error results are obtained, they are sent to the Isolation Analyzer for analysis. Otherwise the intermittent failure message will be repeated. 2. Direct the test to send the error results from the first pass of the failing subsection to the Isolation Analyzer by typing in E(CR). 3. Direct the test to send the results from the second pass of the failing subsection to the Isolation Analyzer by typing in I(CR). 4. Direct the test to resume execution by typing in space(CR). If the repeat subsection bit (PARAM0 bit 2^8) is set, the test will repeat the failing subsection. Otherwise, the test will execute the next subsection.

Message	Explanation
ISOLATION IN PROCESS	The test subsection has detected a hardware failure. The isolate failure bit (PARAM1 bit 2^6) was set and the test has sent the subsection failure data to the Isolation Analyzer for analysis. The Isolation Analyzer will provide the next message informing the CE of the isolation analysis results.
ABORT x	This message, provided by DEX, reports an unrecoverable condition. The test must be reloaded and rerun after the fault condition has been corrected. The various fault values of x are described below: x = 0 Overlay load timeout. 1. Test overlay address destroys DEX. 2. Zero length data generation call to DEX by the test. 3. Test data generation buffer will destroy DEX or the CMSE idler. 4. Test buffers for DEX Compare routine have a zero length specified. 5. Test idled via a D keyboard entry. 6. DEX has been called by the test code with an illegal function (function code greater than 75_8). 7. CMSE did not accept a MCH request/release call.

Error Codes

BASICTC/DEX Error Codes

There are three classes of error codes that will be displayed by the identification tests: internal fault error codes, external fault error codes, and hardware failure error codes. All error codes are found in the EC1 and EC2 portions of the error message.

The internal fault error codes cover all errors that occur within the basic test control code (for example, improper calls to basic test control subroutines from the test section/subsection/condition code). These are unrecoverable errors. The fault condition must be corrected before the test is rerun.

The external fault error codes cover all errors that occur with the external interfaces between the basic test control code and CMSE or maintenance channel, for example, maintenance channel never goes full, bad channel data parity, and so forth. These are unrecoverable errors. The fault must be corrected before the test is rerun.

The hardware failure error codes indicate what part of the hardware being tested has failed. Refer to individual test error codes.

Program Internal Fault Error Codes

EC1 is always equal F170 for internal faults.

EC2 values and meaning are:

Error Code	Meaning
7001	Improper call of basic test control subroutine CMCH.
7002	Improper call of basic test control subroutine CMRW.
7003	Improper call of basic test control subroutine CCSB.
7004	Test subsection has more than allowable 16 conditions.
7005	Revision date of test's microcode binary does not match the revision specified in the PP binary. Note that this is an error generated by the test during initialization and not by Basic Test Control or DEX code. It is a safeguard against the PP and microcode not matching.

External Fault Error Codes

EC1 is always equal 0670 for external faults.

EC2 values and meaning are:

Error Code	Meaning
6001	MCH active when it should not be. CMFN subroutine.
6002	MCH channel error on function. CMFN subroutine.
6003	MCH active when it should not be. CMBT subroutine.
6004	MCH not empty when it should be. CMBT subroutine.
6005	MCH channel error on control bytes. CMBT subroutine.
6006	MCH active when it should not be. CMRD subroutine.
6007	MCH channel error on input. CMRD subroutine.
6010	MCH active when it should not be. CMWT subroutine.
6011	MCH not empty when it should be. CMWT subroutine.
6012	MCH channel error on output. CMWT subroutine.
6013	MCH inactive when it should be active. CMBT subroutine.
6014	MCH inactive when it should be active. CMWT subroutine.
6015	MCH inactive when it should be active. CMRD subroutine.

6050	CMSE call rejected. CALL subroutine.
6052	CMSE program load could not find program on library. CALL subroutine.

Error Code	Meaning
6053	An error occurred on a CMSE program load. CALL subroutine.
6054	CMSE auto PP assignment request could not be honored because all available PPs were already assigned.
6057	CMSE is using CM for communication. Processor tests require all of CM and will not run when CMSE is using CM.
6060	Test could not initialize properly. This is probably due to a hardware failure in an area of logic that the test assumed was functional. To obtain isolation information for this fault, all diagnostics should be run in their proper sequence.
6070	Error in parameter entry. One or more of the test parameters is not consistent with allowable values.

Test Error Codes

Error Code	Meaning/Expected Results
EC1=wxyz, EC2=xxyy	EC1 provides a code that is translated by the test in the error display. It tells what portion of the hardware was being tested when the failure was detected. EC2=xxyy, where xx is the failing section number, yy is the failing subsection number. This information is also provided elsewhere in the tests' error display. For detailed information about the failing section/subsection, refer to the section/subsection documentation in the MSL test description documentation.

Applications

This section describes unique application procedures for the identification tests.

NOTE

Iteration counts are initialized to zero at the beginning of the test and at each end-of-subsection code.

How to repeat an error:

Always take note of the iteration that failed (I=xxxx in the error display) and the conditions that erred (the ones with an "F" in the error display). This information can be used if the test is restarted after a power down/up sequence.

To loop on an error after an error stop:

1. Set stop on iteration (SSI).
2. Set repeat iteration (SRI).
3. Press space bar twice.

The test will repeat the failing sequence with all conditions included to reproduce the error. If the error was solid, it should fail. Proceed to step 4.

If not solid, clear end-of-iteration-stop (CSI) and let the test loop. If the test does not catch the error (it is intermittent), repeat the subsection to get the error back.

4. Error was solid. To keep on the error, set scope mode (SSM). The test is now in a scope loop. This loop may be too long; if so, go to step 5.
5. Type UP,xx (xx=the PP number in which the test is executing. Set report condition errors (SCE). Space test to first failed condition. Set stop-on-condition (SSC) and repeat condition (SRC). Press space bar twice to cause test to loop once on the condition. Set scope mode (SSM). Test will now loop on the error.

   NOTE

   Close inspection of test description manual is needed to determine which condition to scope. A single condition may not always contain all the necessary functions to reproduce the error. Repeat iteration is the best choice in these cases.

To get back to an error after a power down/up sequence (used only on multi-operand/result subsections):

1. Go to the original failing subsection.
2. Set repeat subsection (SRB).
3. Enter PARAM3 (location 125_8) with the desired iteration count (originally obtained from the error display I=xxxx and condition number).

Use EPxx,125,yyyx (yyy=iteration, x=condition number).

4 Press space bar.

When the desired iteration and condition are found, the test automatically sets end-of-condition stop. At this point the set repeat condition (SRC) can be set to loop on the error.

Three choices are now available:

1. Repeat condition.
2. Repeat and use scope mode.
3. Repeat iteration.
   a. Clear stop-on-condition (CSC).
   b. Set repeat iteration (SRI)
   c. Press space bar.

   Test will now repeat the failing iteration.

Detecting and isolating intermittent failures:

To detect an intermittent hardware failure and use the isolation analyzer (available with CMT3 only) to isolate the failure, use the PARAM18 (location 144_8) repeat subsection count +1 parameter; rather than the PARAM0 repeat test or PARAM2 repeat test count parameters.

The first subsection designed to test a particular piece of hardware is more likely to detect intermittently failing hardware when the repeat subsection count parameter is used; resulting in a higher probability of obtaining accurate isolation data from the isolation analyzer.

Appendixes

Glossary A-1

Glossary A

A

Abs

Absolute.

AC or A/C

Address Control.

ACMIC

Address Control Functional Unit Micrand field (bits 64-79).

ADATB/RDSB/B

Refers to B Operand data in Multiply/Divide unit test MDT3.

ADATC/RDSC/C

Refers to C Operand data in Multiply/Divide unit test MDT3.

Adder

Refers to Large Adder in ALN test ANT3. Refer to Multiply Final Adder in test MDT3.

Adder Bit 47 FF

Large Adder Result bit 47 Flip Flop in ALN test ANT3.

Adrs

Address.

ALN

Arithmetic and Logical Network.

ALU

Arithmetic and Logical Unit.

AOR

Address Out of Range.

Arith

Arithmetic.

ASCII

American Standard Code for Information Interchange.

ASE

Address Specification Error.

ASID

Active Segment Identifier.

Assy
Assembly.

Aux
Auxiliary.

B

B
In ALN test ANT3, refers to the operand input to ALN on the RDSB data path.

BCD
Binary Coded Decimal.

BD
Branch Delta.

BDP
Business Data Processor.

Bfr
Buffer.

Bin
Binary.

Bkpt
Breakpoint.

BN
Byte Number.

Bound
Boundary.

Br
Branch.

C

C
In ALN test ANT3, refers to the operand input to the ALN on the RDSC data path.

CA
Condition Action.

CABR
Cache Address Buffer Register.

CAE
Condition Action Extension.

CAR
Cache Address Register.

CBP
Code Base Pointer.

CCR
Clock Condition Register.

CEL
Corrected Error Log.

C.GT.B
C Greater than B signal from the SKG in ALN test ANT3.

Chan
Channel.

Clk
Clock.

Clr
Clear.

CM
Central Memory.

CMC
Central Memory Control.

Cnt
Count.

Cntr
Counter.

Coef
Coefficient.

Com
Common.

Comp
Complement.

Cond
Condition.

Confl
Conflict.

Cont
Control.

Conv
Convert, Converter.

Cor
Corrected.

CPU
Central Processing Unit.

CS
Condition Sensing.

CSA
Control Store Address.

CST
Control Store.

CW1
Control Word 1.

CW2
Control Word 2.

D

DAI
Data Interchange.

DCDR
Decoder.

DCI
Data control information. Two bytes of information that give the maintenance register address.

DEC
Dependent Environment Control, Decimal.

Decr
Decrement.

Descr
Descriptor.

Destn
Destination.

Disassy
Disassembly.

Div

Divide.

Divisor Rgtr

Divide network rgtr internal to LSI CI divide arrays.

Dlyd

Delayed.

DMR

Debug Mask Register.

Dsbl

Disable.

E

EBCDIC

Extended Binary Coded Decimal Interchange Code.

ECC

Error Correction Code.

ECR

C170 Exit Mode Condition Register.

EI

Environment Interface.

EMMR or EMR

Exit Mode Mask Register.

Enbl

Enable.

Encdr

Encoder.

ESE

Environment Specification Error.

EXCH

Exchange.

Exp

Exponent.

Exponent Aux

Exponent Auxiliary board in ALN.

Ext

External.

F

Fctn

Function.

FF

Flip Flop.

FIFO

First In, First Out.

FLC

C170 Mode Central Memory Field Length.

FNO

Fanout.

FRC

Free Running Counter.

FU

Functional Unit.

G

Gen

Generator.

Genl

General.

GME

Used by test CTT3 to indicate a field (bits 62 and 63) within the general micrand.

H

Hldg

Holding.

I

i

Operand Register Designator.

IACY

C170 Parcel Select Mux (IFT3 test).

IAHY

Instruction Assembly Hybrid Mux (IFT3 test).

IAPL

C180 Parcel Select Mux (IFT3 test).

IB02

Instruction Buffer Rank 02 (IFT3 test).

IB12 valid

Used in ALN test ANT3. Instruction Buffer Rank 12 valid signal from Instruction Fetch. This goes active when IF has finished fetching the next instruction from CM.

ICC

Instruction Completion Control.

ICP

Instruction Control Pipeline.

IDS

Immediate Data Select micrand field (bits 26-28).

IDX

Indexed.

IF

Instruction Fetch.

II

Instruction Issue.

Immed

Immediate.

Incr

Increment.

Indef

Indefinite.

Inf

Infinite.

Inh

Inhibit.

Inp

Input.

Instr

Instruction.

Int

Internal.

Integer Mux

Integer/Exponent Mux in ALN.

Intrpt

Interrupt.

IOU

Input/Output Unit.

ISE

Instruction Specification Error.

IU

Instruction Unit.

I

j

Operand Register Designator.

J

JPS

Job Process State.

K

k

Operand Register Designator.

K

170 18-Bit Immediate Operand.

K/L

Key/Lock.

Kypt

Keypoint.

L

l

BDP Operand Length.

Ld

Load.

LM

Local Memory.

LMMIC

Local Memory Functional Unit Micrand field (bits 80-95).

LOS

Loss of Significance.

LRR

Used in ALN test ANT3 and Multiply/Divide
test MDT3, and ICT3 to indicate a live register read.

LRU

Least Recently Used.

LRW

Used in ALN test ANT3 and Multiply/Divide test MDT3 and ICT3 to indicate a live register write.

LSD

Least Significant Digit.

LSI

Large Scale Integration.

LSM

Last Symbol Mask.

LSMPRE

LOAD/STORE Multiple Address Adder.

Lwr

Lower.

M

MAC

Maintenance Access Control.

Maint

Maintenance.

MAR

Micrand Address Register.

MBE

Multiple Bit Error.

MC

Master Clear.

MCH

Maintenance Channel.

MCMR

Monitor Condition Mask Register. Also see MMR.

MCR

Monitor Condition Register.

Mem

Memory.

Micr

Micrand.

Micrand Constant

Contents of ALN FU Micrand bits 0-6.

MIS

Used by tests CTT3 and ICT3 to identify a field (bits 14 and 15) within the general micrand.

Misc

Miscellaneous.

MMR

Monitor Mask Register.

MOP

Micro-operation.

MPS

Monitor Process State.

MR

Maintenance Register.

MSB

Most Significant Bit.

MSC

Micrand Sequence Control. A field (bits 0-15) within the general micrand.

MSD

Most Significant Digit.

MSNZB

Most Significant Non-Zero Byte.

Mult

Multiply.

Mux

Multiplexer.

N

Neg
Negative.

Norm
Normalize.

NPGK
New P Global Key.

NPLK
New P Local Key.

ns
Nanosecond.

O

o
BDP Operand Address.

Op
Operand.

Opcode
Operation Code.

OPGK
Old P Global Key.

OPLK
Old P Local Key.

OPI
Operand Issue.

Opn
Operation.

OSB
Operating System Bounds.

Out
Output.

Ovfl
Overflow.

P

P

Program Address.

Par

Parity.

PDM

Processor Detected Malfunction.

PE

Parity Error.

PFA

Page Frame Address.

PFS

Processor Fault Status.

Ph

Phase.

PIT

Process Interval Timer.

PMF

Performance Monitoring Facility.

PN

Page Number.

PO

Page Offset.

PONR

Point of No Return.

Pos

Positive.

PP

Peripheral Processor.

Prev

Previous.

Pri

Priority.

Prod

Product.

PSM

Page Size Mask.

PSWF

Page Table Search Without Find.

PTA

Page Table Address.

PTD

Page Table Descriptor.

PTL

Page Table Length.

PTM

Processor Test Mode.

PVA

Process Virtual Address.

PW

Partial Write.

Q

Q

C180 16-Bit Immediate Operand.

Quad

Quadrant.

Quotient Rgtr

Divide network rgtr internal to the LSI CI arrays.

R

R/W

Read/Write.

RAC

C170 Mode Central Memory Reference Address.

RAM

Random Access Memory.

RCL

Read and Clear Lock.

Rcvr

Receiver.

RDS

Register/Data Select.

RDSA

In ALN test ANT3, refers to the micrand field that provides the register file address on a write of ALN data and to the data path through DAI for that data.

RDSB

In ALN test ANT3, refers to the following data path to ALN. The specified register file is output to the RDSB bus, transferred to the ALN on ADATB, and is passed through the multiple/divide unit and output Mux to the general network.

RDSC

In ALN test ANT3, refers to following data path. The specified register file is sent to the RDSC bus, transfered to ADATC, and then goes to the general network.

Ref

Reference.

Regen

Regenerator.

Rel

Relative.

Rem

Remainder.

Req

Request.

Resync

Resynchronize, Resynchronizer, Resynchronization.

RF

Register File.

RFOx

Used in ALN test ANT3 and Multiply/Divide test MDT3 to identify the register file being used. x is 5, 6, 7, 8, 9, A, B, C, D, E or F (Hexadecimal).

Rgtr

Register.

RMA

Real Memory Address.

ROM

Read Only Memory.

RSL

Read and Set Lock.

RS64
Right Shift 64 signal from the SKG.

Rxx
Identifies ranks in IF and ICP in Test CTT3.

S

SBE
Single Bit Error.

SC
Soft Control.

SCM
Soft Control Memory.

SCT
Special Character Table.

SECDED
Single Error Correction/Double Error Detection.

Seg
Segment.

Sel
Select.

Sep
Separate.

Seq
Sequence.

Shift Mux
Shift Network Multiplexer.

Shifter
Shift Network.

Sig
Significant.

SIT
System Interval Timer.

SKG
Shift Count Generator.

SM
Segment Map.

SMMIC

Segment Map Functional Unit Micrand field (bits 96-111).

SMO(x)

Segment Descriptor Mux Output bit x.

SMPDM

Segment Map Processor Detected Malfunction.

SN

Segment Number.

Spec

Specification.

SPID

Segment/Page Identifier.

STA

Segment Table Address.

STL

Segment Table Length.

Str

Stream.

Subtr

Subtract.

Suppr

Suppression.

SV

Specification Value.

SVA

System Virtual Address.

Sw

Switch.

Syn

Syndrome.

Sync

Synchronize.

T

Termn

Terminate.

TFA

Tag File Address.

U

u bit

Micrand Bit.

UCEL1

Uncorrected Error Log 1.

UCEL2

Uncorrected Error Log 2.

UCR

User Condition Register.

UMR

User Mask Register.

Unbr

Unbranch.

Uncond

Unconditional.

Uncor

Uncorrectable.

Undfl-

Underflow.

Unlog

Unlogged.

Upr

Upper.

usec

Microsecond.

UTP

Untranslatable Pointer.

Ux

Used in Multiply/Divide test MDT3 to identify Major Cycle Control bits 0-14.

V

VMID

Virtual Machine Identifier.

Vx

Used in Multiply/Divide test MDT3 to identify Minor Cycle Control bits 0-15.

W

WDS

Write Data Select.

WOI

Word of Interest.

X

Xfer

Transfer.

Xltr

Translator.

Xmtr

Transmitter.

XOR

Exclusive OR.

Z

ZIF

Zero Insertion Force.

ZF

Zero Flag.

Comments (continued from other side)

Please fold on dotted line;
seal edges with tape only.

FOLD

FOLD

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL
First-Class Mail Permit No. 8241 Minneapolis, MN

POSTAGE WILL BE PAID BY ADDRESSEE

CONTROL DATA
Technical Publications
ARH219
4201 N. Lexington Avenue
Arden Hills, MN 55126-9983

CIP 960 and 962 Maintenance Software **60000292 A**

We would like your comments on this manual to help us improve it. Please take a few minutes to fill out this form.

Who are you?	**How do you use this manual?**
☐ Manager	☐ As an overview
☐ Systems analyst or programmer	☐ To learn the product or system
☐ Applications programmer	☐ For comprehensive reference
☐ Operator	☐ For quick look-up
☐ Other ______	☐ Other ______

What programming languages do you use? ______

How do you like this manual? Answer the questions that apply.

Yes	Somewhat	No	
☐	☐	☐	Does it tell you what you need to know about the topic?
☐	☐	☐	Is the technical information accurate?
☐	☐	☐	Is it easy to understand?
☐	☐	☐	Is the order of topics logical?
☐	☐	☐	Can you easily find what you want?
☐	☐	☐	Are there enough examples?
☐	☐	☐	Are the examples helpful? (☐ Too simple? ☐ Too complex?)
☐	☐	☐	Do the illustrations help you?
☐	☐	☐	Is the manual easy to read (print size, page layout, and so on)?
☐	☐	☐	Do you use this manual frequently?

Comments? If applicable, note page and paragraph. Use other side if needed.

Check here if you want a reply: ☐

Name	Company
Address	Date
	Phone

Please send program listing and output if applicable to your comment.

CORPORATE HEADQUARTERS, P.O. BOX O, MINNEAPOLIS, MINN 55440
SALES OFFICES AND SERVICE CENTERS IN MAJOR CITIES THROUGHOUT THE WORLD

LITHO IN U.S.A.

CONTROL DATA